

UNIVERSIDADE DE LISBOA  
FACULDADE DE CIÊNCIAS  
DEPARTAMENTO DE INFORMÁTICA



## **Desenvolvimento Aplicacional CRM em Modelo Cloud**

Joana Correia Pinto

**Mestrado em Engenharia Informática**  
Sistemas de Informação

Trabalho de Projeto orientado por:  
Professora Doutora Maria Teresa Caeiro Chambel  
Engenheiro Nuno Infante Alves



# Agradecimentos

Primeiramente, agradecer aos meus dois orientadores: Professora Teresa Chambel e Engenheiro Nuno Infante Alves, pelos conselhos e por sempre me terem ajudado quando precisei. Obrigada aos dois. Um especial agradecimento à minha faculdade - Faculdade de Ciências da Universidade de Lisboa – que foi, praticamente, a minha primeira casa ao longo destes cinco anos e que me proporcionou todas as condições e ambiente para poder concluir os meus estudos.

De seguida, agradecer aos meus pais e irmã por terem acreditado mais em mim do que eu própria. Um especial obrigado à minha irmã Carina, se não fosse pela conversa que teve comigo, provavelmente não teria concluído a minha Licenciatura e, muito menos, o Mestrado.

De seguida, e não menos importante, a todos os meus amigos e família de Faculdade. Não só por terem feito destes últimos anos os melhores da minha vida, mas também pela força que me deram em perseguir o sonho de concluir este curso. E, neste capítulo, tenho de referir os meus dois Padrinhos de Faculdade, Lolada e Ferro, bem como a Marta, Mafalda e Faneca. Sempre me deram apoio e a força que, muitas vezes, estava a escassear. E a todos os meus amigos que me acompanharam nesta jornada, nomeadamente: Branca de Neve, Valeta, Maias, Desmarques, GPS e Gémeas. Foram muitas noites de estudo e de projetos e de muita entreajuda ao longo destes anos todos. Não posso também deixar de referir todos os meus afilhados de Faculdade que, não só tornaram toda esta experiência muito mais enriquecedora, mas também tinham sempre uma palavra de reconforto, motivação e orgulho para me dar enquanto trabalhava para o Mestrado e para o relatório. A todos eles, o meu sincero obrigado pela força e pelas memórias que me proporcionaram.

Aos meus colegas de Mestrado, que tornaram esta nova (mas familiar) jornada muito mais ultrapassável: Branca de Neve, Eunice e Pedro. Foram muitos projetos feitos juntos, com muitas noites em branco, mas sempre com um excelente espírito de camaradagem. Tornaram as coisas muito mais fáceis de ultrapassar e gerir. A vocês, o meu agradecimento.

Relativamente à equipa que estive comigo no meu estágio do Mestrado, um obrigado não chega. Por tudo o que me ensinaram, por todas as dúvidas que me esclareceram, por estarem sempre bem-dispostos e por terem sido os melhores colegas que podia ter tido. Um obrigado

ao João, ao Pedro e ao Tiago. Todos me ensinaram coisas diferentes e, se não fosse por eles, não teria tido a oportunidade que estou a ter agora. Foram os meus mentores e por isso, sou grata. E, para concluir, um obrigado à empresa que me acolheu – Unipartner – por me ter dado as condições necessárias para concluir o meu estágio e por me ter dado as ferramentas necessárias para poder ingressar no mundo do trabalho com capacidades melhores.

# Resumo

O presente relatório descreve o trabalho realizado no Projeto de Engenharia Informática na empresa *Unipartner IT Services*, que decorreu desde novembro de 2016. Este trabalho está inserido no âmbito do projeto “Desenvolvimento Aplicacional sobre Modelos *Cloud*”. Uma solução CRM tem como principais objetivos melhorar as relações entre as empresas e os seus clientes, conquistar a fidelidade dos seus clientes (maximizando as suas relações a longo prazo) e aumentar a rentabilidade – aumentando a taxa de retorno do investimento (ROI), assegurando a competitividade da empresa no mercado.

O projeto dividiu-se em duas fases: uma primeira fase de aprendizagem sobre os conceitos necessários para a elaboração do projeto, de integração necessária na equipa e de familiarização com as ferramentas a serem utilizadas; a segunda fase correspondeu ao desenvolvimento e implementação do projeto. Neste relatório, ambas as fases serão detalhadas.

O projeto consiste na migração de uma solução de CRM (*Customer Relationship Management*) que visa fornecer ao cliente todas as funcionalidades e informação que ele já possuía na solução anterior, com melhorias a nível de desempenho de alguns dos processos e funcionalidades. A solução final desta migração foi construída sobre a versão online do *software* Microsoft *Dynamics 365* (a tecnologia central de todo o projeto), mas que integra com diferentes tipos de outras tecnologias, tais como *Power BI* e Microsoft Azure.

Este projeto surge porque o cliente detetou problemas a nível do tamanho e crescimento da base de dados e do desempenho da sua solução CRM. Estes problemas afetam o serviço prestado pelo cliente aos seus próprios clientes, o que levou à necessidade de reestruturar a arquitetura da presente solução, de forma a colmatar os problemas detetados.

Com a conclusão do projeto, o cliente beneficiou de uma grande melhoria no desempenho do sistema graças à nova forma de armazenamento dos dados: na *cloud*. Praticamente todos os tipos de dados deixaram de estar alocados nos seus servidores *On-Premise* para passarem a estar num repositório Azure. Este fator aliado a uma reformulação das interfaces (tornando-as mais “limpas” e *user-friendly*), a novas maneiras de enviar comunicações aos clientes e de um

novo sistema de suporte ao cliente, fizeram com que o projeto final cumprisse com os objetivos inicialmente delineados pela equipa.

A nível pessoal, o projeto permitiu adquirir conhecimentos técnicos sobre uma tecnologia nova, mas também permitiu adquirir competências sociais: comunicação com o cliente, trabalho de equipa, gestão de tarefas e de tempo e compreensão do mundo empresarial.

**Palavras-chave:** *Customer Relationship Management (CRM)*; *cloud*; migração; desempenho; base de dados

# Abstract

This report describes the work developed in an Informatic Engineering Project in the company **Unipartner IT Services**, which ran since November 2016. This work falls within the scope of the project "Application Development on *Cloud Models*". A CRM solution has as main objectives to improve the relationship between the companies and their clients, conquer their fidelity (maximizing their relationships along the years) and to improve their profitability - improving their return of investment (ROI), ensuring the market's competitiveness.

The project had two phases: one first phase of learning about the required concepts to elaborate the project and familiarization with the tools and technologies that will be used; the second phase corresponded to the development of the project itself. Both phases will be detailed in this thesis.

The project consists on the migration of a CRM (Customer Relationship Management) solution with the goal to provide to the client all the functionalities and information that they already possessed. The final solution of the migration was built under an online version of the Microsoft D365 *software* (the core technology of the whole project), but it integrated with different types of other technologies, such as Power BI and Microsoft Azure.

This project arises because the client, which this project is intended, has detected problems at the level of size and growth of the CRM's database and performance. These problems affect the service provided by the client to their own clients, which led to the need of rebuilding the solution's architecture, to solve the problems detected.

With the completion of the project, the client benefited from a major improvement in system performance, thanks to the new data storage solution: cloud. Practically, almost all data types were no longer allocated on their On-Premise servers, to be on an Azure repository. This factor, combined with an interface reformulation (transforming it more "clean" and user-friendly), to new ways of sending communications to the clients and a new client support system, have made the final project fulfill the objectives initially outlined by the team.

At a personal level, the project allowed to acquire technical knowledge, about a new technology, but it also allowed to acquire social skills: communication with the client, team work, task and time management and knowledge over the business world.

**Keywords:** Customer Relationship Management (CRM); cloud; migration; performance; database



# Conteúdo

<b>Resumo</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Lista de Figuras</b>	<b>ix</b>
<b>Lista de Acrónimos</b>	<b>xi</b>
<b>Capítulo 1 Introdução</b>	<b>1</b>
1.1 Motivação	1
1.2 Objetivos	2
1.3 Instituição de Acolhimento	5
1.4 Planeamento e execução	7
1.5 Organização do Documento	8
<b>Capítulo 2 Customer Relationship Management</b>	<b>11</b>
2.1 Conceitos	11
2.2.3 <i>Enterprise Resources Planning</i>	12
2.2.4 Computação na <i>cloud</i>	12
2.2.5 <i>Customer Relationship Management</i>	17
2.2.6 <i>Cloud CRM</i>	25
2.2.7 Sistema de Extrair, Transformar e Carregar Dados	26
2.2 Trabalho Relacionado	27
2.3 Ferramentas Utilizadas	28
2.3.1 Archi – Ferramenta de modelação	28
2.3.2 Balsamiq – Aplicação de elaboração de <i>mockups</i>	29
2.3.3 Visio – Ferramenta de Desenho de Processos	29
2.3.4 Azure – Serviços <i>cloud</i>	30
2.3.5 Dynamics 365 – Sistema de CRM	31
2.3.6 Microsoft Dynamics CRM 2013 – Sistema de CRM	31
2.3.7 Microsoft Visual Studio 2013/2015 – IDE	31
2.3.8 Plugin Registration Tool – Ferramenta de suporte ao CRM	32
2.3.9 Team Foundation Server – Controlador de versões	32

2.3.10 XrmToolBox – Plataforma de Plugins para o CRM .....	33
<b>Capítulo 3 Implementações em CRM e Migração de Dados e Funcionalidades .....</b>	<b>35</b>
3.1 Fase de Integração com o CRM .....	36
3.1.1 Estudo dos Conceitos – Fase de Integração com o CRM .....	36
3.1.2 Reestruturação da Solução Atual de CRM: Desenho Funcional .....	36
3.1.3 Implementação de Funcionalidades em <i>Javascript</i> e <i>JQuery</i> .....	38
3.2 Desenvolvimentos: Migração de CRM On-Premises para Dynamics 365 na <i>Cloud</i> .....	42
3.2.1 Migração da Solução <i>Business To Consumer</i> (B2C) .....	43
3.4.2 Migração da Solução de <i>Business To Business</i> (B2B) .....	66
<b>Capítulo 4. Conclusão .....</b>	<b>77</b>
4.1 Principais Contribuições .....	77
4.2 Competências Adquiridas .....	78
4.3 Principais Dificuldades .....	80
4.4 Trabalho Futuro .....	81
<b>Referências Bibliográficas .....</b>	<b>83</b>
<b>Anexos .....</b>	<b>87</b>
Anexo I. Organização da Equipa de Desenvolvimento de CRM (para B2C e B2B) .....	87
Anexo II. <i>Archi</i> : Exemplo de Utilização no Projeto .....	89
Anexo III. Formulários Desenvolvidos para o Negócio B2C .....	91
Anexo IV. Menu de Criação e Configuração de um Formulário .....	97
Anexo V. Construção de Regra de Negócio .....	99
Anexo VI. Construção de Fluxo de Trabalho .....	101
Anexo VII. Formulários Desenvolvidos para o Negócio B2B .....	103

# Lista de Figuras

Figura 1.1 - Modelo/Arquitetura do Projeto .....	4
Figura 1.2 - Modelo Organizacional da Unipartner .....	5
Figura 1.3 - Planeamento do Projeto.....	7
Figura 1.4 - Plano de Execução de PEI.....	8
Figura 2.1 - Relação entre os Conceitos Principais do Projeto.....	11
Figura 2.2 - Principais Serviços da Cloud [9].....	14
Figura 2.3 - Sistemas de CRM: Módulos Funcionais .....	19
Figura 2.4 - Registos da Entidade “Contacto” representada no CRM.....	20
Figura 2.5 - Hierarquia das Unidades de Negócio no CRM.....	22
Figura 2.6 - Fluxo do Processo de Negócio no CRM: Exemplo .....	24
Figura 2.7 - Vantagens de uma Solução de CRM na Cloud [5] .....	25
Figura 2.8 - Ilustração de um processo Extrair, Transformar e Carregar .....	27
Figura 2.9 - Desenho de Processos de Negócio no Documento Funcional .....	29
Figura 2.10 - Plugin do XrmToolBox.....	33
Figura 3.1 - Archi: Exemplo de Utilização no Desenho do Processo.....	38
Figura 3.2 - Criação de um Campo Preenchido Dinamicamente em Javascript.....	40
Figura 3.3 - Desenvolvimentos, em JavaScript, no CRM.....	41
Figura 3.4 - Visão Alto Nível da Solução B2C .....	45
Figura 3.5 - Desenho da Entidade “Detalhes de Pagamento” .....	47
Figura 3.6 - Mockup de Formulário do Fã no Documento Técnico .....	48
Figura 3.7 - Componentes Desenvolvidas no B2C .....	49
Figura 3.8 - Menu de Personalização do Formulário do Ticket .....	52
Figura 3.9 - Menu de Propriedades do Formulário.....	53
Figura 3.10 - Seleção do Controlo (Propriedades do Formulário).....	54
Figura 3.11 - Seleção do Evento da IU (Propriedades do Formulário) .....	54
Figura 3.12 - Definição do Campo Calculado “Nº de dias de Resolução” .....	56
Figura 3.13 - Código JavaScript desenvolvido para o Customer Care .....	57
Figura 3.14 - Construção do Fluxo de Trabalho .....	59
Figura 3.15 - Acesso às Aplicações de Negócios.....	61
Figura 3.16 - Estruturador de Aplicações de Negócio .....	62
Figura 3.17 - Mapa do Site do Módulo de Marketing .....	63

Figura 3.18 - Aplicação de Negócio de Marketing.....	63
Figura 3.19 - Visão de Alto Nível da solução B2B .....	67
Figura 3.20 - Componentes Desenvolvidas no B2B .....	68
Figura 3.21 - Extração do Modelo de Dados da Entidade “Canal” do CRM .....	73
Figura 3.22 - Mapeamento do Modelo de Dados entre Solução Antiga e Nova .....	74
Figura 3.23 - Mapeamento do Modelo de Dados entre Solução Antiga e Nova .....	74
Figura 3.24 – Documento de Apoio à Migração da Entidade "Proposta" .....	75
Figura 3.25 - Documento de Criação de Queries para Migração da Entidade "Proposta” .....	75

# Lista de Acrónimos

<b>Acrónimo</b>	<b>Significado</b>
API	<i>Application Programming Interface</i>
BPW	<i>Business Process Flow</i>
B2C	<i>Business To Consumer</i>
B2B	<i>Business To Business</i>
CRM	<i>Customer Relationship Management</i>
DW	<i>Data-Warehouse</i>
D365	<i>Dynamics 365</i>
ERP	<i>Enterprise Resources Planning</i>
ETL	<i>Extract, Transform and Load</i>
PEI	Projeto de Engenharia Informática
SOAP	<i>Simple Object Access Protocol</i>
SSIS	<i>SQL Server Integration Services</i>
SSMS	<i>SQL Server Management Studio</i>
SQL	<i>Structured Query Language</i>



# Capítulo 1

## Introdução

O Projeto de Engenharia Informática (PEI) referente ao presente relatório encontra-se inserido no âmbito do Desenvolvimento Aplicacional sobre Modelos *Cloud*, mais especificamente sobre o ambiente de *Customer Relationship Management* – CRM - a decorrer num ambiente empresarial, na empresa Unipartner. Neste capítulo será referida a motivação para este projeto, os objetivos que foram definidos *à priori* e a organização deste relatório<sup>1</sup>.

### 1.1 Motivação

Nos dias que correm, são muitos os desafios diários que as empresas enfrentam na definição de estratégias que garantam uma gestão eficaz e personalizada dos seus próprios clientes. Também a concorrência existente, por parte de outras empresas, obriga a que a empresa siga uma estratégia de diferenciação e destaque, usando uma abordagem *Customer-centric* na qual os clientes se sintam valorizados e únicos. Nesse sentido, torna-se necessário o desenvolvimento e implementação de soluções que permitam esta gestão do relacionamento das empresas com os seus próprios clientes – *Customer Relationship Management* (CRM), promovendo o conhecimento dos clientes para prestar um melhor serviço e incrementar os resultados da empresa.

Não sendo o CRM uma base de dados (apesar de ter uma base de dados associada), um dos obstáculos diários que as empresas de hoje em dia se deparam, ao usar um sistema de CRM, é o aumento exponencial de informação gerada e a incapacidade do CRM em armazenar grandes quantidades de dados. As empresas reconhecem o grande número de oportunidades disponibilizado por este elevado volume de dados, bem como o seu acrescido valor, nomeadamente em relação ao aumento de vendas e

---

<sup>1</sup> A partir deste ponto, toda a menção feita ao cliente do projeto será feita sob o nome de “Organização”. Todas as referências feitas ao Estágio referir-se-ão ao projeto de mestrado – PEI. As referências ao(s) Projeto(s) serão sobre todos os projetos onde houve envolvimento.

à condução da fidelidade dos seus clientes (um dos princípios chave do CRM). Contudo, é necessário que esta informação não prejudique o desempenho das suas soluções. Assim sendo, é necessário elaborar estratégias para mitigar esta situação, de forma a ter a informação disponível, mas sem diminuir os níveis de desempenho da solução.

## 1.2 Objetivos

Na visão da solução a ser entregue ao cliente, o presente projeto tem como finalidade a execução de um *upgrade* e migração aplicacional de uma solução baseada em Microsoft Dynamics CRM 2015 *On-Premises* para um modelo de CRM Online, integrado com uma instalação já existente de Dynamics CRM Marketing. Esta iniciativa enquadra-se numa visão de implementação mais alargada, onde haverá lugar a outras peças aplicacionais, conceitos e tecnologias. Na solução atual do cliente deste projeto, toda a informação relacionada com os seus clientes é guardada no *Data Warehouse* <sup>2</sup>(DW), onde a comunicação é feita diretamente entre o DW e a instância de CRM da Organização. Com os problemas de desempenho identificados, um novo paradigma foi desenhado. Neste paradigma, DW irá desaparecer dando lugar a um *Data Lake* – um repositório de armazenamento que armazena uma grande quantidade de dados não tratados, na sua forma nativa [30]. Com esta nova abordagem, será necessário criar uma nova camada de serviços Azure que integrarão com o Dynamics 365 de forma a possibilitar a comunicação entre a solução e outros serviços necessários para o projeto principal (de notar que estes serviços estão fora do âmbito do contexto do projeto). Todas estas mudanças farão parte do *upgrade* da solução, onde será necessário construir o CRM de forma a poder interagir com estes serviços. No contexto do projeto apenas houve responsabilidade sobre o desenvolvimento da solução de CRM e da migração de dados e funcionalidades da antiga solução para a nova.

Assim sendo, os objetivos desta nova implementação prendem-se sobre os seguintes pontos:

- Em termos funcionais, a nova solução deverá oferecer o mesmo tipo de serviços e funcionalidades que a atual solução oferece;
- Implementação de estratégias de forma a reduzir a quantidade de informação armazenada do lado do CRM;

---

<sup>2</sup> *Warehousing* - Também conhecido como *data warehouse (DW)*, é um sistema usado como o repositório central de dados integrados de uma organização. É usado em *business intelligence* para a análise de dados e relatórios.



- Integração da solução com os serviços da Microsoft Azure;
- Implementação de estratégias de forma a melhorar o desempenho de algumas operações efetuadas pelo CRM, tais como: envio de mensagens eletrónicas e envio de mensagens de texto;
- Navegação mais ágil e eficiente sobre a solução CRM.

De salientar que o projeto não estava terminado aquando o fim do estágio. Sendo assim, todos os aspectos de avaliação por parte do cliente não serão contemplados no relatório visto que o projeto não chegou a essa fase.

Numa visão pessoal, é expectável que no final do projeto seja possível compreender o processo de implementação de soluções *Cloud* CRM, entendendo todos os processos relevantes para o cliente e a sua transposição para uma plataforma tecnológica. Com a participação ativa na equipa de projeto, deverão ser adquiridos conhecimentos de desenho de uma arquitetura empresarial baseada em serviços *Cloud* e numa plataforma de CRM. Com a conclusão da participação, deverá ser possível produzir as evidências necessárias que sustentem as opções funcionais e tecnológicas tomadas ao longo do projeto, bem como a descrição e definição da solução encontrada e como esta responde aos objetivos do projeto. Assim sendo, foram atribuídos os seguintes blocos de responsabilidades e objetivos:

- 01.** Implementação do bloco de *Customer Care* da instância de B2C da solução (posteriormente este conceito será explicado);
- 02.** Desenvolvimento da documentação funcional e técnica da parte do B2C e B2B. No B2C esta responsabilidade foi partilhada com outro membro da equipa; já na componente B2B a responsabilidade foi total;
- 03.** Criação das Aplicações de Negócio em B2C e B2B – responsabilidade partilhada com outro colega;
- 04.** Criação e desenvolvimentos dos formulários, atributos, regras de negócio e recursos *web* da componente de B2C juntamente com outro colega. Na componente B2B a responsabilidade foi total;
- 05.** Desenvolvimento da componente de Gestores de Alertas em B2C – responsabilidade partilhada com outro colega;
- 06.** Migração dos dados da componente de B2B – responsabilidade partilhada com outro colega.

De seguida segue uma ilustração da arquitetura da interligação entre o sistema de CRM e as respetivas camadas:

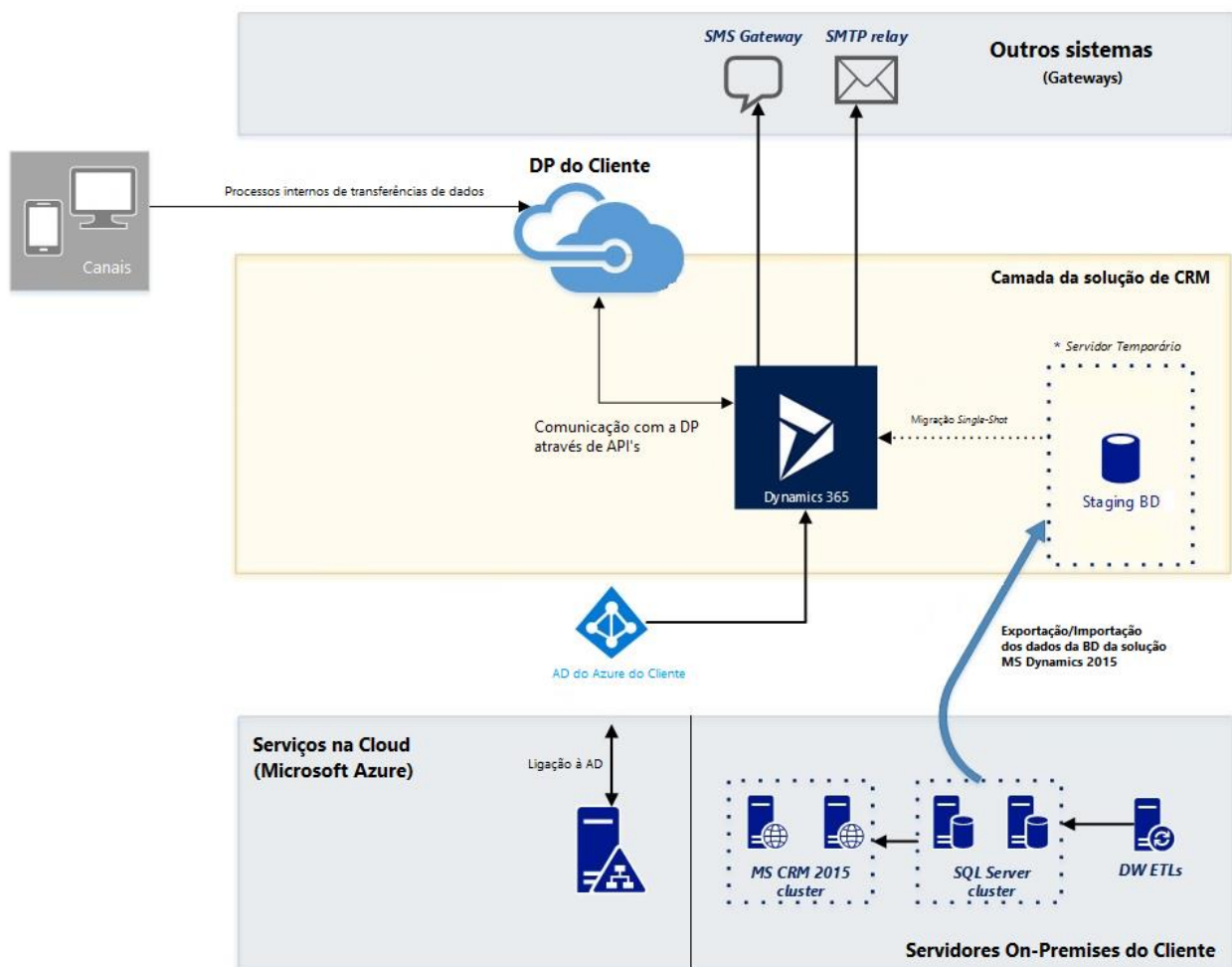


Figura 1.1 - Modelo/Arquitetura do Projeto

Todos os objetivos propostos incidem sobre a camada “**Camada da solução de CRM**”, visto que todos os desenvolvimentos foram sobre a plataforma do CRM. A integração com o *Data Laike* (na figura denominado por “DP do Cliente” (DP significa *Digital Platform*) também teria sido um aspecto a desenvolver, mas devido aos atrasos do projeto, não chegaram a ser concretizados. Alguns desenvolvimentos, nomeadamente na migração de dados, também foram realizados.

### 1.3 Instituição de Acolhimento

A Unipartner é uma empresa de serviços em tecnologias de informação, que trabalha com organizações governamentais, instituições financeiras e empresas comerciais, nos seus desafios de negócio e de IT. É uma empresa privada, criada no seguimento da reorganização da Unisys, em 2015. Na redefinição das operações da Unisys, na EMEA<sup>3</sup> e em Portugal, a Unisys mantém desde então o contacto e o suporte aos seus clientes num modelo indireto, contando com os serviços da Unipartner como *Unisys Service Partner*, assegurando a continuidade dos projetos e relações de negócio, para um vasto número de clientes.

Contando com mais de cem colaboradores no seu arranque, a Unipartner estabeleceu-se como uma empresa nacional líder na consultoria de IT, potenciando uma visão e uma cultura próxima da multinacional Unisys. A oferta da Unipartner inclui a consultoria em transformação digital, implementação de aplicações de negócio, gestão e serviços *Cloud* e de infraestrutura e serviços de outsourcing. A Unipartner tem o reconhecido *EMEA Microsoft Centre of Competence*, em Portugal, entregando soluções inovadoras para as empresas, baseadas nas plataformas Microsoft. Atualmente, o modelo organizacional da Unipartner é o seguinte:

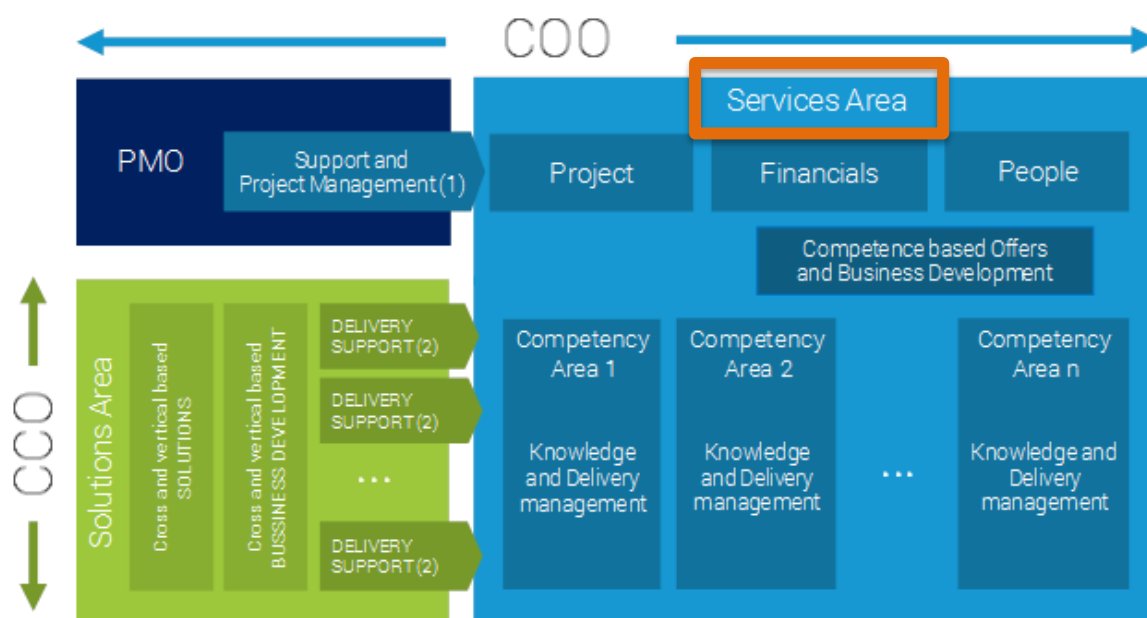


Figura 1.2 - Modelo Organizacional da Unipartner

<sup>3</sup> EMEA significa "Europa, Médio Oriente e África" (*Europe, the Middle East and Africa*) e é uma designação geográfica usada principalmente pelas empresas para descrever os seus negócios na região [12].

Este modelo é constituído por três grandes áreas:

- **Área de Soluções**

Os elementos desta área estarão principalmente focados no *Business Development* de soluções que cruzam várias competências tecnológicas e de soluções verticais.

- É constituída por um conjunto de *Solutions Leaders*, não contendo uma equipa de *delivery*.

- **Área de Serviços (área onde o projeto está inserido)**

Os elementos destas áreas estarão principalmente focados no *delivery*:

- O *Business development* está focado nas competências e *Sell-On*;
- As áreas de competência são geridas por um *Service Leader*.

- **Área PMO**

Esta área tem como função:

- Suportar transversalmente a gestão de projetos;
- Disponibilizar gestores de projeto para determinados projetos.

O presente projeto enquadra-se na área de CRM da empresa (que pertence à área de Serviços, pois a equipa faz a entrega de serviços), que é constituída por dezoito elementos. Todos os elementos estão agrupados em equipas, onde cada uma trabalha para um cliente diferente. Não obstante, cada elemento pode facilmente ser colocado noutro projeto, caso haja essa necessidade. Para o projeto atual, a equipa inicial era formada por quatro elementos: dois elementos dedicados totalmente ao desenvolvimento da nova solução de CRM, um elemento focado na componente da migração dos dados e um sénior a gerir o projeto. Contudo, durante o projeto, a equipa alterou-se (devido à saída de um membro de desenvolvimento), sendo que este foi substituído por um outro elemento. Também o Gestor de Projeto da equipa foi substituído, por volta da mesma altura. O projeto foi desenvolvido nas instalações do cliente, de forma a facilitar a comunicação entre a equipa e o cliente. As constituições das equipas para as duas partes do projeto podem ser visualizadas no Anexo I, no fim do relatório.

## 1.4 Planeamento e execução

Inicialmente, o planeamento do PEI era o seguinte:

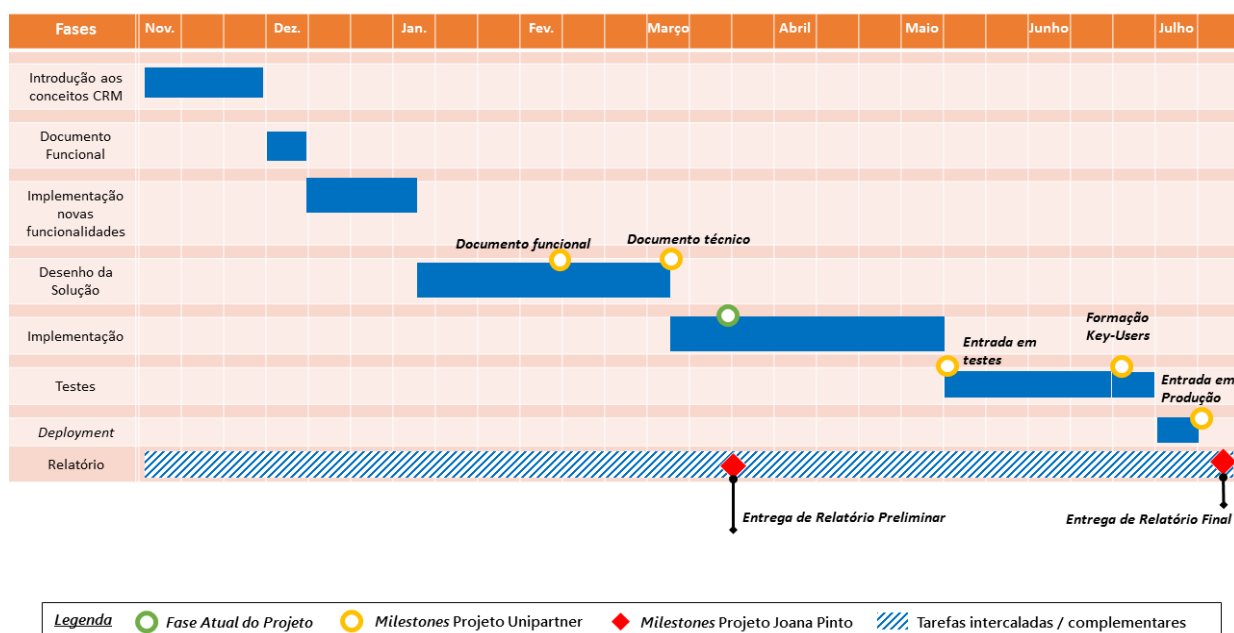


Figura 1.3 - Planeamento do Projeto

Várias situações se alteraram, relativamente ao planeamento inicial. Os documentos funcionais e técnicos foram entregues após a data estimada, o que atrasou o início dos desenvolvimentos. Este atraso aconteceu visto que, parte da documentação, dependia esclarecimentos colocados ao cliente que não foram respondidos dentro do prazo estimado. Adicionalmente, certas seções do documento dependiam da conclusão do desenvolvimento de ferramentas que estavam a ser desenvolvidas por uma outra equipa.

Além disto, durante os desenvolvimentos, o cliente mudou de requisitos fazendo com que a equipa se tivesse de focar, novamente, na documentação e dando menos atenção aos desenvolvimentos em si. Além destes fatores, não foi possível desenvolver certas componentes dentro do período estimado pois havia dependências com uma empresa parceira que deveria ter fornecido o *software* necessário aos desenvolvimentos da equipa de CRM. Como tal não se verificou, esses desenvolvimentos não foram contemplados neste projeto.

Com este atraso significativo, não foi dada a “Entrada em testes”, a “Formação *Key-Users*” e a “Entrada em Produção”, obrigando ao replaneamento do projeto. Sendo assim, a linha de tempo real do que foi executado, foi a seguinte:

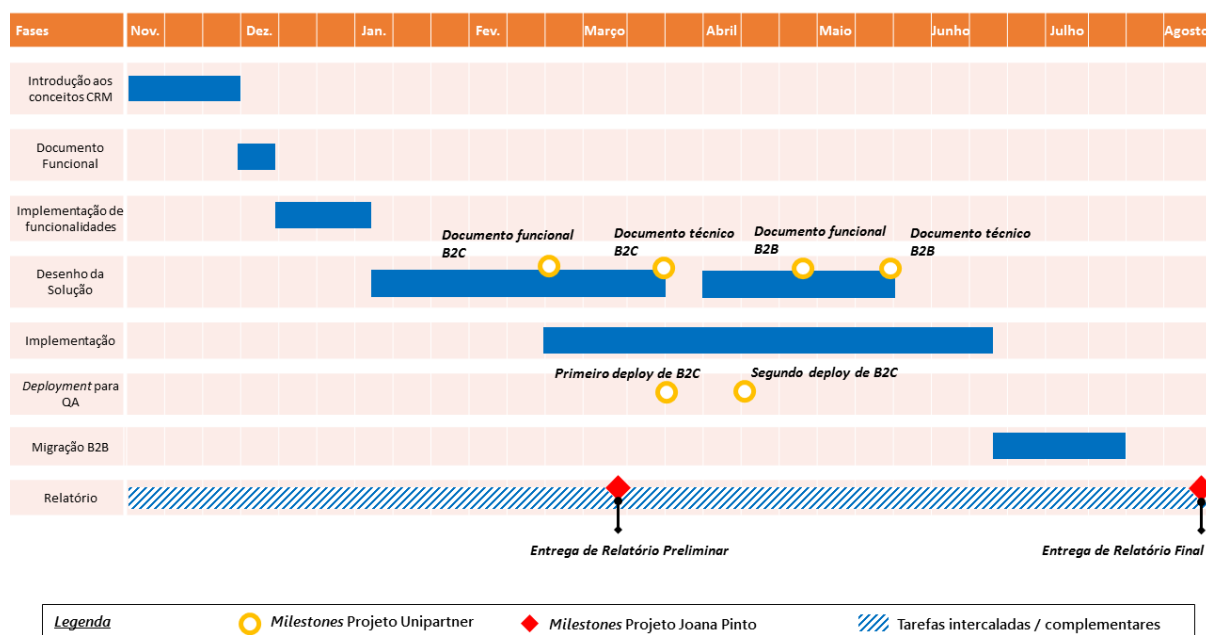


Figura 1.4 - Plano de Execução de PEI

De realçar que o projeto central reportado neste relatório apenas se iniciou em janeiro. Até este mês, outro tipo de trabalho foi realizado em outros dois clientes (mas que apenas tinham o tema do CRM como elo em comum). No primeiro cliente (que pertencia à área de Serviços Energéticos), cujo período de tempo foi de uma semana, foram efetuadas tarefas a nível de documentação funcional onde foram atualizados documentos devido de acordo com os novos requisitos levantados pelo cliente. No segundo cliente (pertencente ao ramo bancário), foram desenvolvidas funcionalidades específicas em JavaScript que manipulavam formulários (interfaces com que os utilizadores interagem) de acordo com os requisitos levantados.

## 1.5 Organização do Documento

O presente relatório encontra-se organizado em quatro grandes capítulos:

- **Capítulo 1**, onde se faz a introdução ao tema do Projeto de Engenharia Informática (PEI). Apresenta-se a motivação para este projeto, os objetivos que foram definidos, a contextualização da empresa e a atual organização do documento;
- **Capítulo 2**, onde se descreve algum do trabalho existente e relacionado com o tema do projeto e também alguns dos conceitos relacionados, de forma a ter uma perceção global do

projeto. Neste capítulo são também descritas as ferramentas utilizadas ao longo de todos os projetos envolvidos;

- **Capítulo 3**, onde se descreve, detalhadamente, todo o trabalho realizado;
- **Capítulo 4**, onde se discutem os pontos mais relevantes do projeto, as principais contribuições do trabalho realizado, as competências adquiridas no decorrer do projeto, bem como as principais dificuldades encontradas e trabalho futuro.





## Capítulo 2

# Customer Relationship Management

Neste capítulo irão ser abordados os conceitos relacionados com o projeto, não só a sua definição, mas como se ligam entre si e se relacionam com a implementação. Também serão referidos alguns trabalhos relacionados com CRM, por parte da Unipartner.

### 2.1 Conceitos

Apesar do conceito principal que envolve este estágio ser o do *Customer Relationship Management*, existem outros que o complementam. Um sistema ERP e um sistema CRM possuem características em comum: a automatização de forças de venda e de *marketing*. São sistemas com algumas similaridades, sendo que um sistema de CRM pode ser, por vezes, considerado um sistema ERP.

Relativamente à computação em nuvem, com o Dynamics 365, este sistema passa a estar na nuvem retirando todas as preocupações de manutenção que um sistema igual, mas On-Premises, traria para o comprador. Assim, pode-se afirmar que o D365 atua como um *Software as a Service*, pois o comprador paga uma subscrição para adquirir este serviço, ficando livre de preocupações de manutenção. A seguinte imagem demonstra como os conceitos abordados neste capítulo se relacionam:

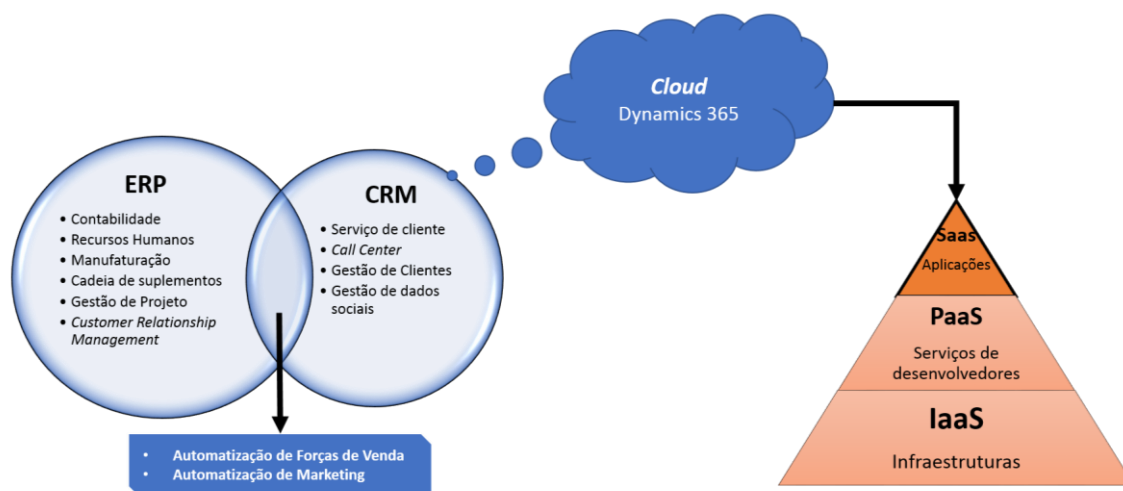


Figura 2.1 - Relação entre os Conceitos Principais do Projeto

De seguida, os principais conceitos serão detalhados e contextualizados com o projeto deste estágio.

### **2.2.3 Enterprise Resources Planning**

Este conceito refere-se a uma gestão integrada de processos de negócio fundamentais, que são aplicados através de tecnologias e *software*. Estas tecnologias e *software* integram os diversos processos e funcionalidades num só sistema de processos *streamline*<sup>4</sup> que são necessários para ter um negócio – inventários, gestão de encomendas, contabilidade, recursos humanos, *Customer Relationship Management*, entre outros [27].

A característica central destes sistemas é a partilha de base de dados que suporta múltiplas funções usadas por diferentes unidades de negócio. Ou seja, utilizadores do sistema das áreas de contabilidade e de vendas, podem usar a mesma informação para as necessidades de cada um.

### **2.2.4 Computação na cloud**

Computação na *cloud* é um conceito que descreve um vasto conjunto de serviços informáticos que são disponibilizados ao cliente, frequentemente, através da Internet. Estes serviços informáticos – servidores, armazenamento, bases de dados, rede, *software*, entre outros – dividem-se em 3 grupos: *Platform as a Service (PaaS)*, *Infrastructure as a Service (IaaS)* e *Software as a Service (SaaS)* [5]. Estes serviços consistem no seguinte:

- ***Software as a Service (SaaS)***

*Software as a Service*, é um modelo onde um *software* é disponibilizado ao consumidor como um serviço cliente (sem a necessidade de haver um formato físico local para o executar – como por exemplo, quando a instalação de um *software* é feita a partir de um CD/DVD) onde um ponto de acesso a esse *software* lhe é fornecido de forma a que fique a executar no dispositivo do cliente [28]. Um bom exemplo para este tipo de serviço é o utilizado para o presente projeto: o D365 que é fornecido ao cliente via *web browser* (através da aquisição de uma subscrição). O uso deste

---

<sup>4</sup> *Streamline* - Processos para ajudar a melhorar a eficiência do negócio através da simplificação, ou eliminação, de passos desnecessários usando técnicas modernizadas ou aplicando outras abordagens [37].

tipo de serviços reduz os custos de aquisição de um *software*, visto que remove a necessidade de ter uma equipa técnica de instalação, manutenção e melhoramento do *software*.

Contudo, este tipo de serviço não é aconselhável para todos os tipos de situações. Devido às limitações no acesso à Internet, este tipo de serviço nem sempre pode corresponder às expectativas. Também quando a empresa segue uma legislação que não permite a hospedagem de dados fora da empresa ou os requisitos de segurança e SLA – representam um “contrato” entre o fornecedor e o cliente que documenta que serviços o fornecedor irá disponibilizar e que define os padrões de desempenho que o serviço necessita de cumprir - são críticos, é aconselhável usar uma abordagem diferente.

- ***Platform as a Service (Paas)***

Tipicamente, este tipo de serviços fornece uma plataforma onde um determinado *software* possa ser desenvolvido e instalado, de forma a abstrair toda a infraestrutura que rodeia um ambiente deste tipo. Para atingir tal efeito, estes serviços são desenvolvidos usando tecnologia de virtualização. Assim os programadores conseguem investir o seu tempo na parte do desenvolvimento e escalabilidade da sua aplicação. Um exemplo deste tipo de serviço é a *Google App*.

- ***Infrastructure as a Service (IaaS)***

*Infrastructure as a Service*, são os serviços base necessários em qualquer plataforma *cloud*: servidores, armazenamento, rede e sistemas operativos. Para uma empresa, isto significa que não precisam de comprar uma infraestrutura completa para o desenvolvimento do seu *software*, bastando adquirir estes recursos, na *cloud*. Além do custo destes recursos ser inferior ao custo da infraestrutura física, também os custos de manutenção deixam de ter valor, sendo uma mais valia para a empresa.

A figura abaixo demonstra o tipo de recursos que cada um destes três serviços disponibiliza, bem como as responsabilidades atribuídas à empresa e ao fornecedor do serviço desses recursos, dependendo do modelo usado. Fica claro que um *SaaS* retira todas as responsabilidades de manutenção ao utilizador, visto que o produto que lhe é vendido está construído de forma a que ele não tenha preocupações e que simplesmente consiga usufruir do *software*. Pelo contrário, um *IaaS*

apenas retira ao utilizador as responsabilidades relativas à infraestrutura física para a construção do *software*. A figura também ilustra o contexto em que todos os serviços estão alocados On-Premises, onde toda a infraestrutura, rede, aplicações de desenvolvimento são da responsabilidade da empresa e/ou utilizador, bem como os custos associados.

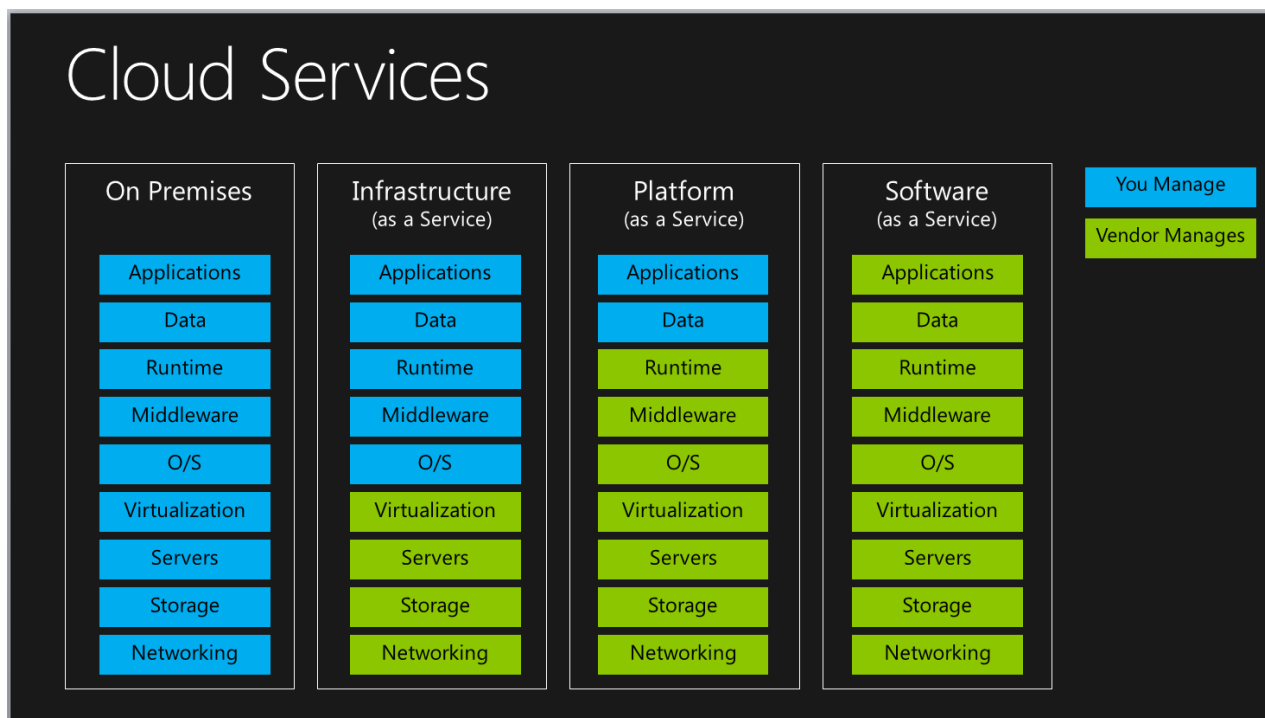


Figura 2.2 - Principais Serviços da Cloud [9]

Hoje em dia, o modelo em *cloud* é **cada vez mais usado pelas organizações** [26], sejam elas pequenas ou grandes, pelos seguintes **motivos**:

### 1. Custo

O menor investimento inicial (CAPEX<sup>5</sup>) associado a este tipo de modelos, que advém do facto de se eliminar os gastos de capital associados à compra de hardware e de *software*, bem como à configuração e execução de *datacenters* (local onde se encontram os equipamentos de processamento e armazenamento de dados de uma empresa ou organização).

<sup>5</sup> CAPEX - Acrónimo para “*Capital expenditure*” que se refere à quantia de dinheiro usado na aquisição, ou introdução de melhorias, de bens de capital de uma determinada empresa [11].

## 2. Velocidade

Grandes quantidades de recursos informáticos podem ser disponibilizadas em poucos minutos, visto que a maior parte destes serviços são disponibilizados como *self-service* e a pedido. Assim, é proporcionada às empresas uma enorme flexibilidade relativamente ao planeamento da capacidade.

## 3. Escalabilidade

Como foi referido anteriormente, todos estes serviços informáticos são disponibilizados através da Internet. Ou seja, é possível fornecer ao cliente os serviços por ele desejados no momento exato em que são pedidos e independentemente da localização geográfica, o que faz com que estes serviços possam ser disponibilizados em várias regiões do mundo e em grande escala.

## 4. Produtividade

A computação na *cloud* permite que as equipas de TI se possam dedicar a objetivos comerciais mais importantes que tarefas como configuração e manutenções de *hardware*, aplicação de *patches* a um certo *software*, entre outras, que seriam necessárias se fosse um modelo *on-premise* (modelo onde o *software* é executado nas máquinas que se encontram nas instalações da própria empresa ou organização).

## 5. Desempenho

Os grandes serviços informáticos funcionam numa rede mundial de *datacenters* seguros, que são atualizados regularmente para a última geração de *hardware* informático rápido e eficiente. Assim, a latência de rede é mais reduzida para aplicações face ao cenário de ter um único *datacenter* empresarial.

## 6. Fiabilidade

Este tipo de modelo torna o sistema muito mais fiável, no sentido em que torna possível armazenar os dados em diversos locais redundantes na rede do fornecedor do *cloud service*. Assim, fica mais barato e fácil, para as empresas e/ou organizações, executar cópias de segurança de dados e recuperá-los após algum desastre no sistema.

Contudo, e apesar da popularidade deste tipo de modelo, existem algumas **debilidades** que se deve ter em conta, antes de se adquirir um serviço deste género [16]:

## **1. Lentidão**

Mesmo tendo uma ligação rápida à Internet, as aplicações baseadas em *web* podem ser de acesso lento, comparativamente com um *software* semelhante executado num *desktop*. Esta lentidão deve-se ao facto de, normalmente, todos os componentes sobre os quais o *software* está a ser executado estarem alocados fisicamente em pontos distantes do globo, provocando latência nas comunicações.

## **2. Recursos limitados**

Nem todas as aplicações baseadas em Internet oferecem um vasto leque de recursos ao utilizador. Comparando os recursos da *Google Presentations* com o Microsoft PowerPoint, torna-se claro a gritante discrepância de funcionalidades que uma e outra aplicação têm.

## **3. Segurança**

Neste modelo, os dados ficam armazenados na *cloud*. Os utilizadores com acesso à *cloud* encontram-se em permanente dúvida relativamente à segurança dos seus dados, pois as questões de segurança e privacidade ainda não são completamente asseguradas pelas companhias que fornecem este tipo de serviço. Contudo, este ponto dependerá sempre do fornecedor do serviço *cloud*.

## **4. Dependência dos dados**

Com os dados armazenados remotamente, caso a *cloud* perca estes dados, serão perdidos para sempre. Apesar de, normalmente, os dados estarem replicados por diversas máquinas, a verdade é que não é um dado garantido e o utilizador está dependente da *cloud* e do seu bom funcionamento.

É necessário analisar, cuidadosamente, se o produto a ser desenvolvido vai beneficiar (ou não) com as vantagens do modelo em *cloud* pois as desvantagens deste modelo podem resultar em problemas críticos para a empresa que o esteja a usar (como a perda total de dados ou o acesso não-autorizado a dados críticos e confidenciais). Sendo assim, num modelo CRM, este é também um ponto crítico visto que a solução pode estar a armazenar informação confidencial e crítica.

### **2.2.5 Customer Relationship Management**

*Customer Relationship Management* (CRM) refere-se a estratégias, tecnologias e práticas que as organizações usam para gerir e analisar os dados e as interações dos clientes, durante o ciclo de vida do cliente. Este tipo de metodologia prende-se com os objetivos de melhorar as relações com os clientes e com potenciais clientes, de aumentar as vendas, de preservar os clientes existentes [31] – através do melhoramento da experiência dos mesmos - e de diminuir os custos da gestão dos clientes.

Estes sistemas são desenhados de forma a ser possível compilar informação sobre os clientes. Não só a informação pessoal do cliente, mas também o seu historial de compras, as suas preferências de compra, as suas reclamações, entre outros [31]. Esta informação pode ter origem em diferentes canais de comunicação: *website* da empresa (caso o cliente seja uma empresa), telefone, e-mail, *live chat*, materiais de *marketing*, redes sociais, fax, entre outros.

Um sistema CRM ajuda ao crescimento do negócio de uma empresa ou organização porque guarda o histórico das interações com o cliente – chamadas, *emails* trocados, compromissos, apresentações, os passos necessários para a conclusão de um negócio ou para prestar um melhor serviço, entre outros. Este histórico torna-se importante para o desenvolvimento da relação entre o cliente e a organização, pois ajuda a reduzir a pesquisa do cliente, dentro do sistema de CRM, e a correlação entre os clientes. Com estes dados recolhidos, torna-se possível antever as necessidades dos clientes de uma forma eficaz e que beneficie o negócio, bem como ter uma previsão sobre a rentabilidade do cliente para a empresa ou organização, ou ainda, para melhorar o serviço prestado aos clientes.

Nos próximos dois pontos serão abordados os conceitos técnicos e funcionais do D365. De notar que serão estas as componentes e técnicas usadas no projeto, sendo que esta próxima descrição é num contexto geral.

#### **2.2.5.1 Conceitos Funcionais**

Neste tópico será abordado, do ponto de vista funcional, a estrutura dos sistemas CRM e o que eles têm a oferecer aos seus utilizadores. Geralmente, os sistemas CRM encontram-se divididos em 3 módulos [36]:

- **Vendas:** este módulo ajuda as equipas de vendas a estruturarem os seus processos comerciais desde a fase de pré-venda até à conclusão da venda. O sistema CRM permite que as interações com o cliente sejam registadas e guardadas. Através da construção de fluxos de trabalho automatizados, é possível alertar os comerciais de possíveis produtos/ofertas que um determinado cliente está propenso a comprar e a gestão das oportunidades comerciais;
- **Marketing:** permite que as equipas de marketing planeiem atividades de marketing relacionadas com o negócio, tanto a longo como a curto prazo [7]. Para conseguirem concretizar estas ações de Marketing, utilizam componentes nativa do CRM:
  - Listas de Marketing: fornecem uma maneira de agrupar contatos e interagir com eles, através de emails, chamadas telefónicas, entre outros. Estas listas permitem a segmentação dos clientes, como por exemplo, permitindo criar uma lista de contactos de maiores de idade;
  - Campanhas: componente que permite medir a eficiência da ação de marketing e que serve de recurso para atingir um determinado resultado, tal como a venda de um produto da empresa. Para tal, escolhe-se um canal de comunicação que é por onde a campanha será distribuída. Por exemplo, através do email;
  - Campanhas Rápidas: muito semelhante à campanha, mas apenas pode ser relacionada a um tipo de atividade (só a email ou só a chamada telefónica, por exemplo).
- **Serviço:** este módulo permite que os utilizadores do CRM consigam prestar um serviço ou dar suporte aos clientes. Assim que um cliente entrar no sistema, a empresa tem de disponibilizar ao cliente todos os serviços de suporte necessários. Desde assistência às suas compras, pagamentos, até à disponibilidade em responder a qualquer tipo de questões colocadas pelo cliente. Todos estes possíveis “problemas” criados pelo cliente, dão origem aos chamados “Incidentes”. Desta forma, as equipas que estiverem a cargo da tramitação destes incidentes, conseguem fornecer um serviço de qualidade na resposta às preocupações do cliente.

A próxima figura representa, muito resumidamente, os módulos funcionais de um sistema de CRM:





Figura 2.3 - Sistemas de CRM: Módulos Funcionais

Estes módulos são, frequentemente, chamados de “Áreas de Trabalho”. Os três podem trabalhar em conjunto, durante o ciclo de vida do negócio, de forma a conquistar novos clientes (através do *Marketing*), a vender-lhes os seus serviços (através das Vendas) e a manter os clientes existentes (através do Serviço).

#### 2.2.5.2 Modelo de dados e Processos de Gestão de Informação

Para que os sistemas CRM conseguissem ajudar as empresas a gerir os seus negócios e clientes, é necessário que existam componentes que suportem diversos tipos de operações, tais como: armazenamento e criação de dados, aplicação de regras de negócio, processos despoletados por eventos, relacionar diferentes dados de informação entre si, entre outros. Para conseguir executar estas operações, o CRM oferece várias componentes, que interagem entre si, e que permitem aos programadores configurar o sistema de acordo com os requisitos do cliente. Seguidamente, irão ser apresentadas as componentes que constituem o CRM e que permitem que o fluxo do negócio se realize.

#### Entidades e atributos

As entidades são tabelas que armazenam um certo tipo de dados, usando os seus atributos como colunas da tabela, determinando que tipo de informação vai ser guardada nos registos da entidade. Ou seja, criar um registo numa entidade no CRM é análogo a criar um registo numa tabela de uma base de dados, onde a entidade corresponde a uma tabela e os atributos da entidade correspondem às colunas da tabela da base de dados [17].

Por definição, o CRM já engloba algumas entidades nativas e atributos. Ou seja, são componentes que são criadas automaticamente assim que uma instância de CRM é criada. Contudo, o utilizador pode alterar estas entidades e atributos nativos se assim o desejar (com algumas limitações, no entanto). Além destas entidades nativas, o utilizador pode criar novas entidades, personalizadas por si assim como os seus respetivos atributos.

A figura abaixo ilustra um exemplo de uma entidade chamada “Contacto” e alguns dos seus atributos (Nome Completo, Correio Eletrónico, Endereço, entre outros):

✓	Nome Completo ↑	Correio Eletrónico	Nome da Empresa	Telefone da Empresa	Proprietário	Endereço 1: Localidade	Endereço 1: País/Região (N...)	Klout Score
	Abraham McCormick	someone29@contoso.com			SYSTEM			
	Adriana Cunha	george@graphicdesigninst...	Graphic Design Institute	111-587-4582	Constância Ribe...	Redmond	US	
	Adriana Lopes	sueburki@marginstravel.com	Margie's Travel	123-879-9905	Lara Batalha (Da...	Newport Beach	US	
	Adrienne McMillan				SYSTEM			

Figura 2.4 – Registos da Entidade “Contacto” representada no CRM

Esta componente é o núcleo de um sistema de CRM, pois todas as operações possíveis ocorrem à volta desta informação armazenada, como irá ser explicado um pouco mais adiante. Além disto, regras de negócio que possam ser exercidas sobre estes dados, irão ser realizadas sob as entidades e os seus atributos.

## Formulários

Para apresentar ao utilizador a informação contida nas entidades, o CRM utiliza uma componente chamada “Formulário” que tem o propósito de apresentar os dados da entidade ao utilizador de forma a que ele possa interagir com ela, como está ilustrado na Figura 2.4 – *Registos da Entidade “Contacto” representada no CRM*, mais abaixo. Nos formulários, a informação apresentada não é só a que está diretamente ligada à entidade em si, mas também informação de registos de outras entidades relacionadas (através das relações existentes entre as entidades – à semelhança do conceito da “relação” de uma base de dados relacional).

O utilizador tem assim o poder de manipular que informação quer apresentar e de que forma a vai apresentar. Com a ajuda de recursos web em JavaScript e HTML, tem também a possibilidade de apresentar dados de forma dinâmica ou estática. Assim, os programadores têm uma maior flexibilidade na construção dos formulários, tendo mais possibilidades ao seu dispor do que as que o sistema de CRM oferece nativamente. Um exemplo de um formulário pode ser visualizado no Anexo III.

De notar que uma entidade pode ter associada a si um ou mais formulários. Esta opção torna-se muito útil quando é necessário dividir os acessos que cada utilizador do sistema de CRM pode ter a um formulário.

### **Relações entre entidades**

Tal como num sistema de base de dados relacional, para haver correspondência entre os diferentes tipos de informação é necessário que duas tabelas tenham um elo de ligação entre si (as chamadas **chaves estrangeiras**). De forma semelhante, nos sistemas CRM, existem **Ligações** que se criam entre duas entidades, através dos seus atributos. Estas ligações podem ser de três tipos:

- **Um-para-muitos (1:N)**: significa que um registo de uma entidade X pode ter vários registos da entidade Y associados a ele. No CRM estas relações são apresentadas no formulário do registo como subgreijas;
- **Muitos-para-um (N:1)**: contrariamente à relação 1:N, nesta relação um registo da entidade X só pode estar associado a um e só um registo da entidade Y. Contudo, o mesmo registo da entidade Y pode estar associado a diferentes registos da entidade X. No CRM, estas relações são apresentadas no formulário do registo como um campo de pesquisa;
- **Muitos-para-muitos (N:N)**: nesta relação, vários registos da entidade X podem estar relacionados com vários registos da entidade Y e vice-versa.

Quando se define o tipo de relação que duas entidades vão ter entre si, também é necessário definir o comportamento destas relações. Ou seja, definir se o comportamento vai ser referencial, em cascata ou parental (tal como existe nas bases de dados relacionais).

### **Unidades de negócio**

As unidades de negócio compõem a fundação da estrutura de segurança do CRM. Cada utilizador do sistema pertence a uma unidade de negócio, obrigatoriamente. Em cada unidade de negócio é possível criar uma ou mais equipas, compostas por diferentes utilizadores. Existem unidades de negócio nativas que não podem ser eliminadas nem desativadas, apenas se pode alterar o seu nome [33]. Contudo, os utilizadores podem criar mais unidades de negócio onde cada uma tem diferentes níveis de acesso a cada entidade do CRM. A imagem seguinte mostra a hierarquia entre as unidades de negócio, as equipas do CRM e os níveis de permissão:

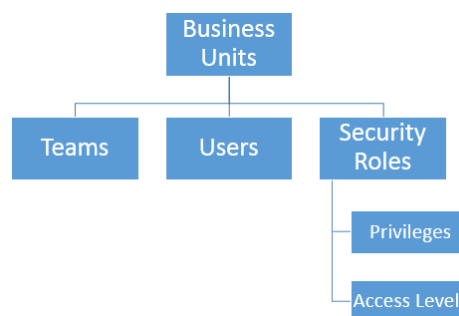


Figura 2.5 - Hierarquia das Unidades de Negócio no CRM

### Fluxo de Trabalho (*Workflow*)

Esta ferramenta do CRM permite criar processos de negócios simples e autónomos. Podem ser criados através das funcionalidades *out-of-the-box*<sup>6</sup> do CRM ou através da implementação por código em .NET [14]. Os fluxos de trabalho podem ser executados em segundo plano ou em tempo real. Podem ou não receber um *input* por parte do utilizador. Eles são iniciados através de condições específicas, definidas pelo utilizador, ou podem ser disparados manualmente.

Para configurar um fluxo de trabalho, têm de ser aplicadas as seguintes configurações:

- Definir sobre que entidade do CRM ele vai ser executado;
- Definir se vai ser executado de modo síncrono ou assíncrono;
- Definir o evento que o vai disparar (na criação, alteração de campos, atribuição, eliminação ou alteração de estado do registo da entidade);

---

<sup>6</sup> *Out-of-the-box* refere-se às funcionalidades ou características que um *software* oferece e que podem ser imediatamente utilizadas após a instalação do *software*, sem necessitar de configurações [15].

- Definir o contexto sobre qual o fluxo de trabalho vai ser executado (ou seja, se o fluxo de trabalho vai ser executado sobre registos possuídos por utilizadores, unidades de negócio ou por toda a organização que compõe o CRM);
- Definir as ações e fases.

Vendo o seguinte exemplo: a empresa pretende enviar um e-mail a um cliente, assim que ele é registado no sistema do CRM. O utilizador teria de criar um fluxo de trabalho de forma a que fosse executado sobre a entidade “Contacto” (em modo assíncrono), onde iria ser disparado no momento da criação do registo, sobre o contexto da organização e com a ação “Enviar Correio Eletrónico”. Desta maneira, a organização consegue ter processos de negócio a correr automaticamente e só os têm de definir uma única vez.

### **Fluxo do Processo de Negócio (*Business Process Flow*)**

Esta ferramenta do CRM tem como objetivo guiar os utilizadores durante um processo de negócio do sistema. É composto por Fases onde em cada Fase existem passos a serem cumpridos, através dos atributos da entidade. No cabeçalho do BPF o utilizador consegue visualizar em que fase está, as próximas fases e o que necessita de realizar para atingir a próxima fase (Figura 2.6 - *Fluxo do Processo de Negócio no CRM: Exemplo*). Como quase tudo no CRM, também estes BPF podem ser personalizados pelos utilizadores e podem ser usados em entidades nativas e customizadas [33]. A grande diferença entre este fluxo e o fluxo de trabalho prende-se no automatismo da tarefa, onde no fluxo do processo de negócio esse automatismo é inexistente pois o objetivo é guiar o utilizador. Na imagem seguinte está representado um fluxo de processo de negócio que contém quatro fases: Identificar, Qualificar, Pesquisar e Resolver. A interface indica ao utilizar a fase em que ele se encontra de momento (neste caso, assinalado através do ícone da bandeira, o utilizador está na fase “Pesquisar”). Para poder prosseguir para a fase seguinte, necessita de ter certos requisitos cumpridos (assinalados a preto). Todos estes passos e definições são feitos pelo utilizador, de forma a criar processos que estejam de acordo com o negócio em que a sua empresa está inserida.

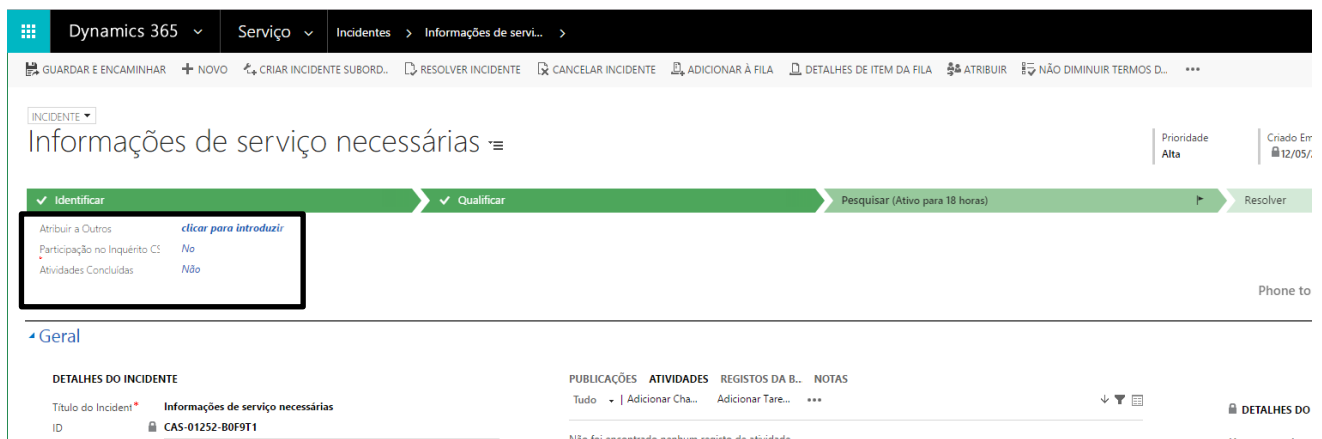


Figura 2.6 - Fluxo do Processo de Negócio no CRM: Exemplo

Todas estas ferramentas são essenciais pois, em conjunto, oferecem às empresas a capacidade de gerir tanto os seus negócios como os seus clientes, de forma amigável e facilitada para os utilizadores, pois não precisam de ter conhecimento técnicos para os criar. Além das capacidades inerentes deste *software* também é possível integrar o D365 com diversos sistemas externos que ajudem ao negócio da empresa e que ofereçam serviços que esta plataforma não possui ou que não possui de maneira tão eficiente.

## Vistas

As Vistas de CRM são listas de registos que respeitam um determinado critério estabelecido pelos utilizadores. Também é possível definir as colunas que se quer apresentar. Imaginando uma lista de contactos de uma empresa, caso quisessem apenas pessoas com idade superior a 18 anos criariam uma lista cujo o critério fosse Idade superior a dezoito apresentando as colunas desejadas. Esta é uma ferramenta extremamente útil na medida em que dá uma grande flexibilidade aos utilizadores pois cada um deles pode criar as suas próprias vistas, criando-as de acordo com as suas necessidades. Além disto, as vistas podem ser utilizadas para questões de segmentação. Por exemplo, na criação de uma Campanha o utilizador pode indicar, no CRM, que quer que essa campanha seja executada sobre uma determinada lista. Esta característica traz agilidade e rapidez neste tipo de processo.

## 2.2.6 Cloud CRM

Juntando os dois conceitos acima detalhados – CRM e Computação na *Cloud* - surge um novo: *Cloud CRM*, que representa todo o *software*, ferramentas de CRM e dados que se encontram alocados numa *cloud*. Este tipo de *software* pode ser acedido através da Internet (geralmente baseado em ferramentas de *web* ou *web browsers*), facilitando os utilizadores na consulta de informação a qualquer altura. Algumas das vantagens que este tipo de modelo oferece estão contempladas na seguinte figura:

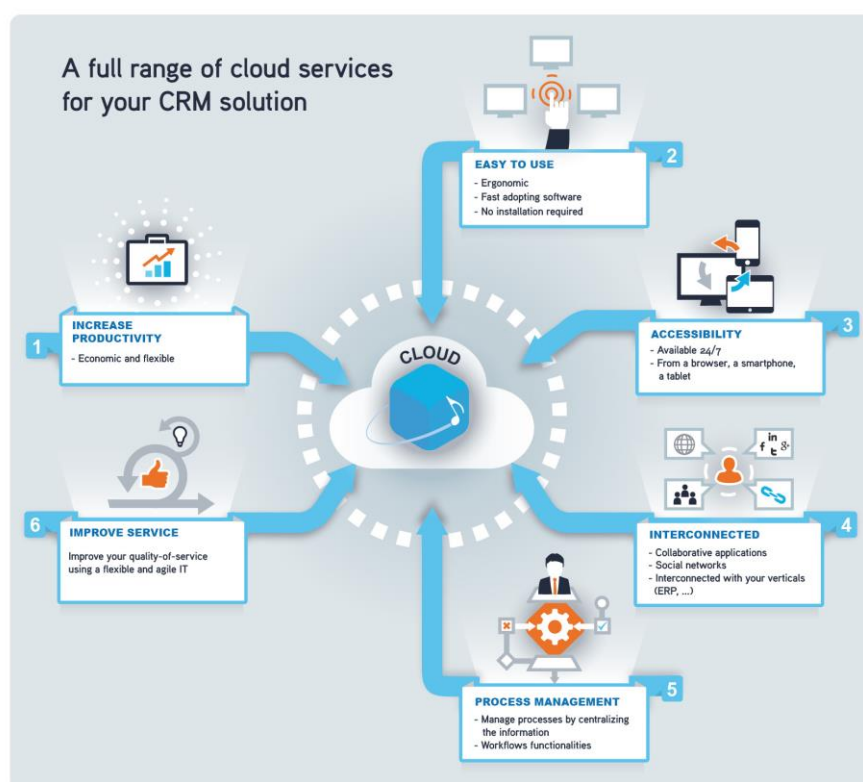


Figura 2.7- Vantagens de uma Solução de CRM na *Cloud* [5]

Este factor aliado à rapidez da *cloud* e à facilidade de executar a instalação, entre outros, tornam este modelo extremamente atrativo a organizações e empresas com recursos ou peritos tecnológicos limitados [31]. Este conceito pertence à categoria **SaaS** (*Software as a Service*).

### 2.2.7 Sistema de Extrair, Transformar e Carregar Dados

ETL, acrónimo para *Extract, Transform and Load* (extrair, transformar e carregar), refere-se a um processo usado nas bases de dados e, especialmente, em *data warehousing*. Insere-se no contexto onde é necessário carregar para o DW os dados de um outro sistema fonte, de forma a facilitar o processo da análise de negócio [13]. Este processo é composto por 3 etapas:

- **Extração dos dados**- os dados do sistema externo ao DW são extraídos e convertidos num formato consolidado de DW, estando assim prontos a serem transformados e mapeados para o formato de dados do sistema alvo da operação. Estes dados serão guardados numa área de *staging*<sup>7</sup> (passo 1 da Figura 2.8).
- **Transformação dos dados**- esta etapa consiste nas transformações feitas aos dados, de forma a que o seu estado final seja o estado dos dados que o sistema alvo suporta (passo 2 da Figura 2.8). Pode envolver as seguintes tarefas:
  - Aplicação de regras de negócio
  - Limpeza
  - Filtragem
  - Divisão de uma coluna em múltiplas colunas, ou vice-versa
  - Junção de dados de várias fontes
  - Transposição de colunas e linhas
  - Aplicação de validações de dados, simples ou complexas
- **Carregamento dos dados**- fase final onde os dados são carregados para o DW (passo 3 da Figura 2.8).

A figura seguinte ilustra todo este processo descrito:

---

<sup>7</sup> *Staging* - Também chamada de *landing zone*, refere-se a uma área intermediária que é usada para armazenar dados que estão a ser processados durante um processo de ETL. Esta área situa-se entre o local de origem e de destino dos dados.



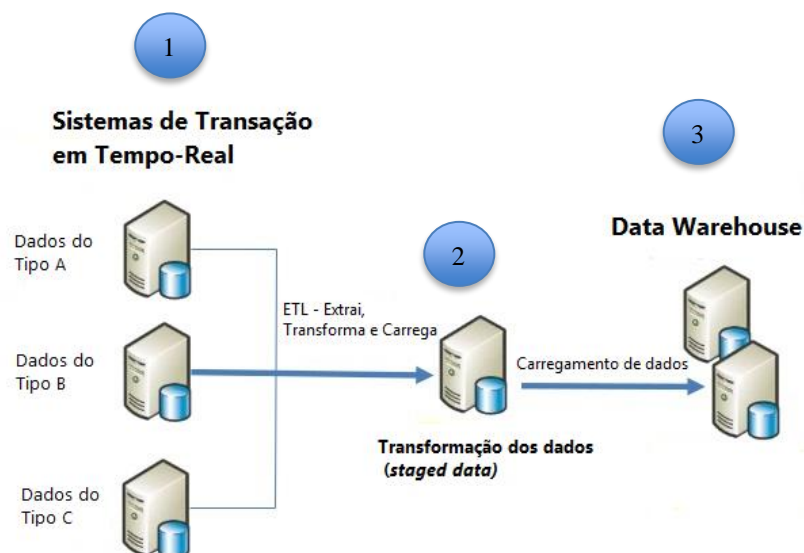


Figura 2.8- Ilustração de um processo Extrair, Transformar e Carregar  
1- Extrair; 2- Transformar; 3 - Carregar

Este conceito de **Extrair, Transformar e Carregar** acompanhou grande parte do projeto, visto que este era um projeto que incluía a migração de funcionalidades e dados.

## 2.2 Trabalho Relacionado

A Unipartner é parceira da Microsoft com a competência **Gold Cloud CRM**, atuando na área do CRM como assessor de negócio, no desenho e na implementação de soluções Dynamics CRM, contando com duas décadas de experiência acumulada, desde que era Unisys.

As soluções CRM promovem normalmente a redução de custos, o aumento das vendas ou a melhoria da qualidade de serviço ao cliente.

A Unipartner tem implementado diversas soluções de CRM em diferentes setores da indústria. No setor público, na administração central por exemplo, com soluções de gestão de reclamações ou de gestão centralizada de entidades. Na administração local, por exemplo com soluções de balcão único de atendimento para as autarquias. No setor financeiro, com soluções de automação da força de vendas, automação de marketing e de gestão de *Compliance*. No setor da educação, com processos

de candidatura e de gestão de bolsas e subsídios, ou a gestão do ciclo de vida do proponente, entre outros.

Com estas soluções, a Unipartner, pretende entregar soluções que acrescentem valor ao negócio dos seus clientes, endereçando as necessidades de negócio ou problemas de forma pragmática e consultiva.

## **2.3 Ferramentas Utilizadas**

Todas as ferramentas utilizadas para os projetos até agora realizados são ferramentas já existentes e que foram usadas para realizar as tarefas pretendidas. As ferramentas foram usadas para os seguintes propósitos do projeto:

- Desenho da solução: criação da documentação funcional e técnica (ponto **2.3.1 - 2.3.3**);
- Desenvolvimento da solução: personalização da solução de CRM (ponto **2.3.4 - 2.3.7**);
- Ferramentas de auxílio ao desenvolvimento do projeto (ponto **2.3.8 - 2.3.10**).

De seguida, será descrita a utilização de cada uma dessas ferramentas.

### **2.3.1 Archi – Ferramenta de modelação**

Ferramenta de modelação para criar modelos e desenhos baseados na linguagem *ArchiMate*. Consegue suportar a descrição, análise e visualização da arquitetura dentro, e transversalmente, dos domínios de negócio [2].

É uma ferramenta direcionada a arquitetos e modeladores empresariais, conseguindo preencher os requisitos de quase todos os arquitetos e os seus *stakeholders*.

Esta ferramenta foi utilizada numa primeira fase do estágio, onde foi necessário desenhar diversos processos de negócio.

### 2.3.2 Balsamiq – Aplicação de elaboração de *mockups*

Balsamiq é uma ferramenta de *wireframing*<sup>8</sup> que permite o desenho de *mockups*, reproduzindo a experiência de desenhar num quadro branco (mas através de um computador) [3]. Para este projeto, foi utilizado para desenhar os *mockups* dos formulários da solução de CRM, de forma a dar ao cliente uma pré-visualização do que seriam os formulários, futuramente.

### 2.3.3 Visio – Ferramenta de Desenho de Processos

Microsoft Visio é uma aplicação gráfica de vetores e diagramas. Dentro dos diagramas, é possível criar diversos tipos: organogramas, fluxogramas, modelagem de dados – usando UML ou outras notações gráficas disponíveis – diagramas de redes, plantas baixas, cartazes, entre outros [23].

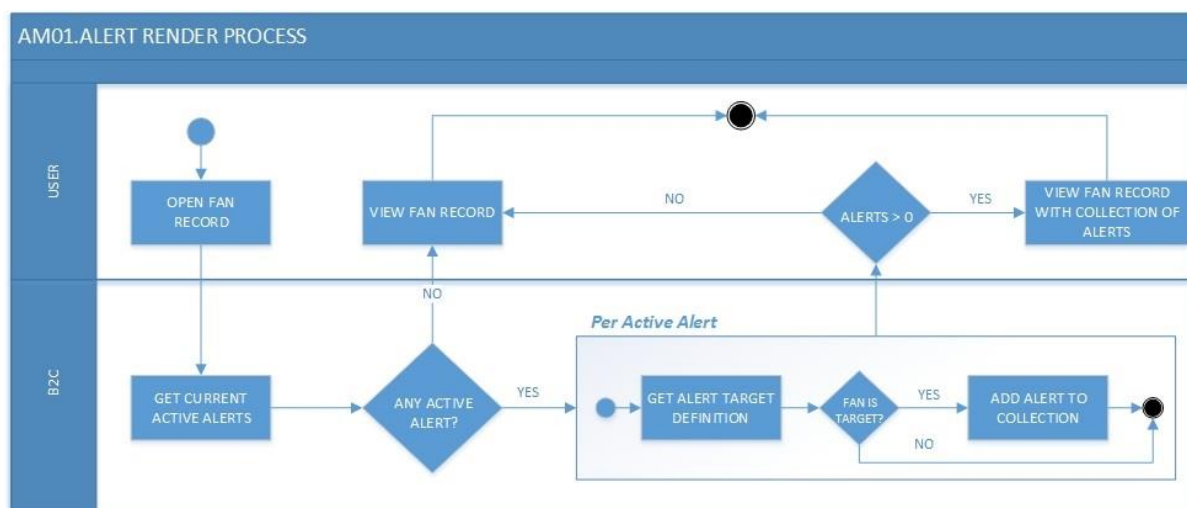


Figura 2.9 - Desenho de Processos de Negócio no Documento Funcional

Além de todas as capacidades acima referidas, esta ferramenta permite a inserção de dados reais (via Excel). O Visio foi usado durante a fase de desenho do projeto atual (Documentos de Desenho Funcional e Técnico).

<sup>8</sup> *Wireframing* refere-se a uma metodologia de desenhar *websites* a um nível estrutural. Assim, permite que sejam desenhados os elementos chave de uma página web [8].

### 2.3.4 Azure – Serviços *cloud*

Microsoft Azure refere-se a um conjunto de serviços - de análise, computação, bases de dados, dispositivos móveis, redes, armazenamento e Web - integrados na *cloud* que permitem aos programadores a criação de aplicações de uma forma mais rápida, intuitiva e fácil (visto que, muitas vezes, abstrai toda a componente de comunicação entre as aplicações). Permite uma movimentação mais rápida e uma maior concretização de objetivos [25]. Consegue suportar um grande conjunto de sistemas operativos, linguagens de programação, arquiteturas, ferramentas, bases de dados e dispositivos. Além disto, torna possível a criação de aplicações em *JavaScript*, *Python*, *.NET*, *PHP*, *Java* e *Node.js*.

No contexto do projeto, foram utilizados duas componentes do Azure: *webjobs* e *Service Bus*. Os *webjobs* são programas que executam tarefas em segundo plano. Podem correr continuamente, a mando do utilizador ou por agendamento.

Por outro lado, o *Service Bus* é um serviço de *cloud* multi-inquilino [14], o que significa que vários utilizadores partilham o serviço, que permite a troca de mensagens entre várias aplicações. Para esta troca de mensagens poder ser feita, existem três mecanismos que o utilizador pode utilizar:

- **Filas**, que permitem comunicação unidirecional. Cada fila funciona como um intermediário (por vezes denominado mediador) que armazena as mensagens enviadas até serem recebidas. Cada mensagem é recebida por um único destinatário.
- **Tópicos**, que proporcionam uma comunicação unidirecional através de subscrições – um só tópico pode ter várias subscrições. Tal como as filas, o tópico funciona como um mediador, mas cada subscrição pode utilizar opcionalmente um filtro para receber apenas as mensagens que correspondem a critérios específicos.
- **Reencaminhamentos**, que proporcionam comunicação bidirecional. Ao contrário das filas e tópicos, o reencaminhamento não armazena as mensagens em trânsito - não é um mediador. Simplesmente, transmite-as à aplicação de destino.

Apesar destes componentes e conceitos terem feito parte do projeto, todos os desenvolvimentos relacionados com eles foram efetuados por um colega de equipa, tendo ficado como responsabilidade apenas uma prova de conceito em que era necessário ligar a solução de CRM à ao Azure e enviar mensagens através de filas.

### 2.3.5 Dynamics 365 – Sistema de CRM

D365 refere-se a um *cloud service* (SaaS), desenvolvido pela Microsoft, que combina os serviços de CRM (*Customer Relationship Management*) e ERP (*Enterprise Resources Planning*) num só serviço e que também inclui aplicações para gerir todo o serviço de vendas, marketing, automatização de serviço de projeto, finanças, operações e serviço do cliente. Esta ferramenta será o núcleo do projeto, visto que toda a solução será construída sob este tipo de sistema.

### 2.3.6 Microsoft Dynamics CRM 2013 – Sistema de CRM

Solução de negócio de CRM desenvolvida pela Microsoft (*On-Premise*), que encapsula todas as boas práticas de CRM com flexibilidade, familiaridade e funcionalidade. Foca-se essencialmente em Vendas, Marketing e Serviço ao Cliente. É uma aplicação de servidor-cliente que suporta SharePoint e extensas interfaces de *Web services*. Pode ser acedido ou através de um *browser* ou através de um *plugin* no Microsoft Outlook. Esta versão do CRM foi utilizada num projeto intermediário (anterior ao projeto principal do estágio) pois era a solução atual da empresa em questão.

### 2.3.7 Microsoft Visual Studio 2013/2015 – IDE

O Microsoft Visual Studio é um IDE (*Integrated Development Environment*) que é usado para desenvolver programas computacionais, *websites*, aplicações *web* e serviços *web*. Inclui um editor de código que suporta *IntelliSense*<sup>9</sup> assim como *code refactoring* [24] e também uma componente de *debugging* que trabalha tanto a nível do código como da máquina, baseada na colocação de *breakpoints*<sup>10</sup>.

É uma ferramenta que suporta diversas linguagens de programação: C, C++, C++/CLI, VB.NET, C#, F#, *Python*, Ruby, Node.js, XML/XSLT, HTML/XHTML, JavaScript, CSS, Java, entre outras.

---

<sup>9</sup> *IntelliSense* é uma implementação da Microsoft de *code completion*, que é uma característica existente em alguns ambientes de programação que ajudam a acelerar o processo do ato de programar.

<sup>10</sup> *Breakpoints* permitem que a execução do código seja parada, temporariamente, numa certa posição do código.

O *Visual Studio* foi utilizado durante a fase de implementação de novas funcionalidades, mas também foi utilizado no projeto central para a criação de recursos web da solução e para os processos de migração. *SQL Server 2014 Management Studio*.

O *SQL Server Management Studio* é um ambiente integrado para gerir infraestruturas de SQL, desde servidores SQL a base de dados SQL [19]. Este *software* fornece ferramentas para configurar, monitorizar e administrar instâncias de SQL que tenham sido implementadas. Este *software* foi usado para a componente de migração do projeto, onde foram criadas as tabelas, os campos, os *procedures* <sup>11</sup>necessários para executar as *queries* criadas.

### 2.3.8 Plugin Registration Tool – Ferramenta de suporte ao CRM

Ferramenta que utiliza uma interface gráfica para dar suporte ao registo de *plugins* e de atividades de fluxo de trabalho customizadas no servidor da solução de CRM, de forma a que estes possam ser utilizados convenientemente na solução desenvolvida. Esta ferramenta vem no pacote SDK do Microsoft Dynamics CRM.

### 2.3.9 Team Foundation Server – Controlador de versões

Geralmente abreviado de TFS, o Team Foundation Server é um *software*, desenvolvido pela Microsoft, que permite a gestão de código fonte, relatórios, requisitos, projeto (tanto para desenvolvimento de *software* em *agile* como em cascata), *builds* automatizadas, testes e de capacidades de gestão de lançamento de *software*- inclui todo o processo de gestão, planeamento, calendarização e de controlo do desenvolvimento de *software* durante as diferentes fases e ambientes (tais como as de teste e de *deployment*) [18].

Falando mais propriamente do desenvolvimento de *software*, o TFS fornece diversas ferramentas de desenvolvimento de *software* colaborativo que integra com o IDE, ou editor de código, existente no utilizador. Assim, permite que uma equipa multifuncional consiga trabalhar de uma forma eficiente num projeto de grande, média ou pequena dimensão.

---

<sup>11</sup> *Procedure* refere-se a um subprograma que executa uma ação específica.

Graças à sua capacidade de planeamento e estimação de tempos das tarefas, esta ferramenta está atualmente a ser utilizada para gerir o tempo e as tarefas atribuídas a cada membro da equipa.

### 2.3.10 XrmToolBox – Plataforma de Plugins para o CRM

XrmToolBox é uma aplicação Windows que se liga a uma aplicação CRM e que disponibiliza ferramentas, ao utilizador, que lhe ajudarão a personalizar e configurar tarefas mais facilmente, como por exemplo: visualização dos recursos web da aplicação CRM, alteração de dados de registos em massa, processamento de fluxos de trabalho em massa, entre outros.

É constituída por trinta e um *plugins*, até à última versão publicada (v.1.2017.2.13), mas qualquer pessoa pode criar um *plugin* e adicionar ao XrmToolBox, bastando implementar uma interface [34].

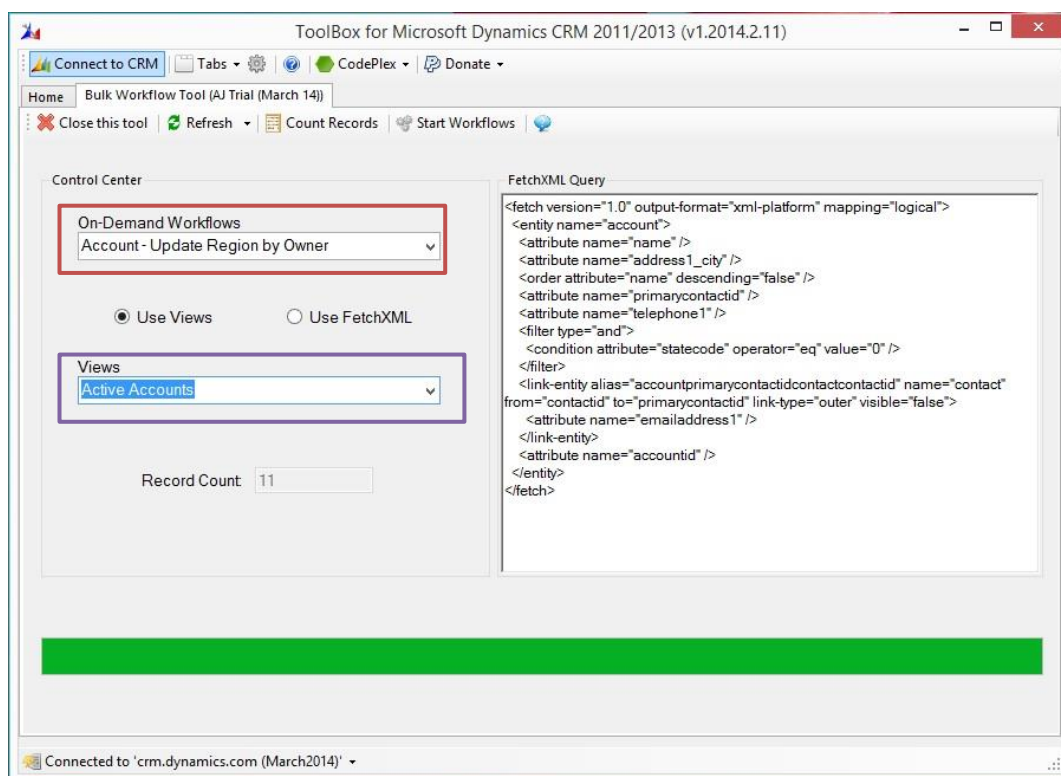


Figura 2.10- Plugin do XrmToolBox  
Processamento de um Fluxo de Trabalho em Massa

A figura acima demonstra um exemplo de uma das funcionalidades desta ferramenta. Neste caso, o utilizador vai executar um fluxo de trabalho em massa (ou seja, sobre um grande número de registos) onde o campo “*Region*” da entidade “*Account*” será atualizado. Para executar este processo, o utilizador tem de escolher o fluxo de trabalho que quer executar (assinalado pela caixa a vermelho

na imagem de acima). Seguidamente, tem de seleccionar se pretende acionar o fluxo de trabalho sobre uma Vista (escolhendo na *dropdown* a vista que deseja, assinalado pela caixa de cor roxa da imagem acima) ou usando *FetchXML*<sup>12</sup>.

---

<sup>12</sup> *FetchXML* é uma linguagem usada para criar *queries* sobre a base de dados do Microsoft D365 [19].



## Capítulo 3

### Implementações em CRM e Migração de Dados e Funcionalidades

Neste capítulo, o foco será sobre todos os desenvolvimentos práticos efetuados ao longo do estágio. Como já foi referido no início deste relatório, o estágio foi dividido em duas fases: a fase de aprendizagem dos conceitos e de integração na equipa (que será designada **Fase de Integração com o CRM**); e a fase do desenvolvimento do projeto principal (que será designada **Fase de Desenvolvimento**). Na próxima secção iremos abordar a primeira fase.

Durante os primeiros dias do estágio, não houve a participação de um projeto em si. O objetivo inicial era a interiorização dos conceitos envolvidos na plataforma que iria fazer parte do projeto do estágio: o D365. Esta fase ocorreu no escritório da Unipartner, juntamente com a ajuda de um consultor sénior, que seria o consultor responsável pelo primeiro projeto real que iria ser realizado.

Concluída esta fase, um novo projeto se iniciou e desta vez no local onde o cliente estava alocado. Este projeto fazia parte de uma empresa da área de serviços energético, que pretendia reestruturar a sua solução atual de CRM visto que esta teria de ser integrada com um novo Portal de Queixas nacional. Sendo assim foi necessário analisar os novos requisitos da solução e redesenhá-la. Este período durou somente uma semana e teve como objetivo ganhar experiência na parte de Análise de um projeto, visto que durante essa semana, o trabalho debruçou-se todo sobre a construção do documento funcional da nova solução.

Terminada a semana, iniciou-se novamente um novo desafio. Desta vez, a empresa era um banco nacional e que precisava de implementar novas funcionalidades na sua solução atual do CRM. Este período foi mais longo, durou cerca de um mês, e foi bastante diferente da experiência anterior visto que o projeto estava inserido numa fase diferente: numa fase de implementação. O objetivo deste período foi ganhar conhecimento técnico sobre o CRM pois, até ao momento, todos os conceitos abordados tinham sido a nível funcional.

Por fim, iniciou-se o projeto central deste estágio que foi começado na fase de desenho da solução, visto que o levantamento de requisitos já tinha sido efetuado pela equipa. Este projeto continuou até o fim do estágio e englobou os dois tipos de experiência adquiridos nas fases anteriores: desenho

da solução (que inclui a criação do documento funcional e técnico da solução) assim como a sua implementação e fase de testes.

Os subcapítulos seguintes aprofundarão as fases mencionadas anteriormente, dando uma visão mais detalhada sobre o que foi feito e como foi feito.

### **3.1 Fase de Integração com o CRM**

A fase de integração com o CRM consistiu no período de adaptação e de aprendizagem, relativamente aos sistemas de CRM e a todos os conceitos relacionados com eles. Para conseguir concretizar o pretendido, foram necessários dias de pesquisa, de visualização de formações online e de algumas reuniões com consultores séniores que não só ajudaram na compreensão dos conceitos, mas também realizaram exercícios de teste, de forma a entender se o conhecimento estava ou não a ser interiorizado. E depois, a integração em projetos reais e com tarefas específicas, de forma a não só pôr em prática todos os conceitos aprendidos, mas também a ganhar conhecimento mais técnico sobre os sistemas CRM.

#### **3.1.1 Estudo dos Conceitos – Fase de Integração com o CRM**

Inicialmente, o que se fez foi um estudo dos conceitos inerentes ao CRM: o que é, quais as suas componentes, como utilizar o Microsoft Dynamics CRM, entre outros. Após este estudo, foram propostos exercícios que consistiam em desenhar a solução para um determinado problema. Concluindo os exercícios, onde houve um debate com um consultor sénior da empresa sobre os problemas encontrados, bem como o porquê das escolhas tomadas para obter aquela solução, seguiu-se uma nova etapa. Toda esta etapa foi realizada no escritório da Unipartner.

#### **3.1.2 Reestruturação da Solução Atual de CRM: Desenho Funcional**

Antes de começar a participação no novo projeto, foi dado um certo período de tempo – cerca de duas semanas - para a sua contextualização onde ocorreram reuniões com o consultor sénior, responsável pelo projeto, de modo a que pudesse dar uma visão geral do projeto. Este novo projeto consistia na reestruturação de uma solução CRM já existente, por parte de uma certa empresa que era encarregue de atender queixas de clientes da área de serviços energéticos. Esta solução passaria

a integrar com um novo portal de queixas, externo à solução, (que tinha sido desenvolvido e colocado em funcionamento) e que faria com que a solução mudasse ligeiramente a sua arquitetura. A participação neste projeto foi feita nas instalações do cliente.

Nesta nova etapa, e já com a participação num projeto concreto, o trabalho realizado debruçou-se sobre o desenho funcional do sistema a ser implementado, visto que se encontrava na fase de análise do projeto. Com o levantamento dos requisitos do cliente já efetuados anteriormente pelo consultor sénior responsável, estes foram mapeados, analisados e descritos com detalhe no documento do desenho funcional. Foi necessário não só elencar os requisitos, mas também ligá-los aos componentes que constituíam o sistema – tanto os que já existiam como os novos - bem como aos objetivos a serem alcançados. Esta ligação foi realizada com uma ferramenta de modelação denominada “*Archi*”, (referida em 2.3.1) que permite criar modelos e desenhos baseados na linguagem “*Archimate*”. No Anexo II é possível visualizar um dos desenhos realizados com uma visão global sobre toda a solução. Esta linguagem suporta a descrição, análise e visualização da arquitetura do sistema em questão, abrangendo os diversos domínios do negócio. Um exemplo de utilização desta ferramenta encontra-se na Figura 3.1 - *Archi: Exemplo de Utilização no Desenho do Processo*, onde se pode verificar os diferentes requisitos do sistema ligados a serviços e componentes do sistema. Isto permite ter uma rastreabilidade da solução, no sentido em que se um requisito falhar, rapidamente, a equipa responsável saberá que serviços e componentes irão ser afetados, conseguindo assim prevenir e mitigar potenciais problemas de uma maneira mais eficaz e rápida.

Complementarmente à listagem dos requisitos do sistema, foi também necessário desenhar os processos inerentes ao sistema, de uma forma compreensível aos olhos do cliente. Este desenho foi concretizado usando, novamente, a ferramenta “*Archi*”. De notar que todos os desenhos concretizados foram complementados com uma pequena legenda de forma a familiarizar o cliente com os conceitos da linguagem “*Archimate*”. A imagem seguinte reflete um destes processos desenhados onde, neste caso, se tratava do processo de consulta dos detalhes de um pedido de reclamação feito por um cliente:

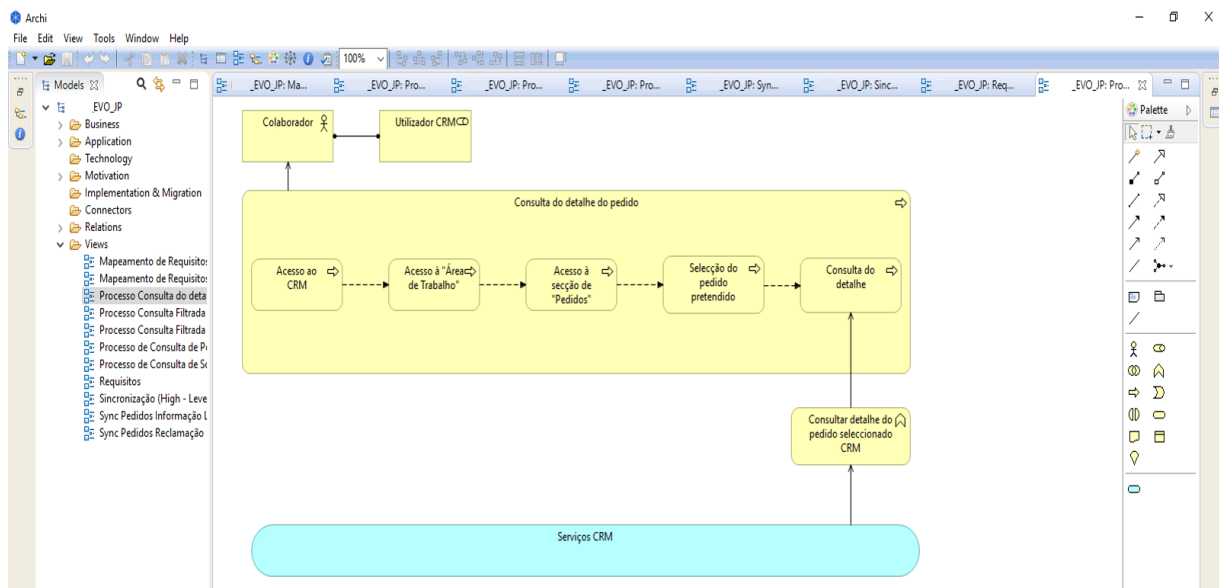


Figura 3.1 - Archi: Exemplo de Utilização no Desenho do Processo

Finalmente, após a elaboração dos passos supramencionados e da devida análise e validação por parte do Gestor de Projeto da equipa, a próxima e última fase seria a elaboração dos casos de uso do sistema, usando as interfaces do sistema. Contudo, esta participação no projeto foi interrompida, visto que se avançou para um novo projeto pois era necessário ter uma pessoa adicional a ajudar num outro projeto. Como tal, não se chegou a desenhar estes casos de uso, de maneira que não existem exemplos para exibir.

Os maiores desafios nesta fase do estágio foram conseguir entrar na lógica de negócio num espaço de uma semana e ter a capacidade de contribuir para a documentação funcional de forma rápida e eficiente, tendo em conta o curto período de tempo que este projeto durou. Contudo, a integração na equipa e no projeto foi bem-sucedida, tendo contribuído para a criação de esquemas e de desenhos de processos do projeto.

### 3.1.3 Implementação de Funcionalidades em *Javascript* e *JQuery*

Nesta etapa, o tipo de trabalho a realizar mudou bastante. Anteriormente, o trabalho estava numa fase de análise. Focou-se, em grande parte, na recolha de requisitos e na sua listagem, no desenho da solução, na listagem das componentes do sistema, ou seja, focou-se muito na parte documental do projeto. Enquanto que nesta etapa, o trabalho já estava numa fase de implementação e focou-se na parte do desenvolvimento de novas funcionalidades, sobre um sistema de CRM do cliente, já completamente funcional e operacional. Para contextualizar, o sistema referido pertence a um

sistema bancário nacional. Este sistema continha, entre outros, dados dos clientes, dados de produtos que lhes poderiam ser vendidos, interações que foram feitas entre o cliente e o comercial da agência bancária (chamadas telefônicas, trocas de *mails*, entre outros).

Foram usadas duas linguagens de programação diferentes para a implementação destas funcionalidades: *HTML5* e *JavaScript* (adicionalmente, foi usada a biblioteca de *JQuery*). Além de trabalhar com estas linguagens, as próprias funcionalidades permitidas pelo sistema do CRM foram usadas para atingir os objetivos propostos.

Relativamente às funcionalidades implementadas, temos a seguinte lista:

- a.** Criação de uma nova secção do CRM, de forma dinâmica e respeitando certos critérios, relativos à sua visibilidade;
- b.** Criação de novos campos, apresentados na secção acima referida, de forma dinâmica;
- c.** Adaptação de campos já existentes com as novas funcionalidades implementadas;
- d.** Validações sobre campos associados a vendas de produtos, efetuando acessos e operações sobre a base de dados do CRM;
- e.** Apresentação de dados de forma dinâmica e com validações relativamente à visibilidade, ou não, destes dados.

A grande dificuldade sobre a implementação destas funcionalidades foi a adaptação inicial a um projeto já existente e com um elevado grau de complexidade. Além de diversas operações possíveis, existiam muitas entidades, muitos campos que exigiam um nível de conhecimento e perceção elevados. Também era importante conhecer as relações que existiam entre as diferentes entidades e entender o que já existia no código original e que poderia ser útil para estas novas implementações. Estas preocupações serviram para reduzir o risco de duplicar funcionalidades e de ter código extra, sobre um sistema já extremamente carregado de funcionalidades e de código.

Na Figura 3.2 - *Criação de um Campo Preenchido Dinamicamente em Javascript*, a coluna “Venda com Sucesso” foi uma das funcionalidades implementadas. Este campo permitia que os utilizadores soubessem se uma determinada oferta da empresa já tinha sido vendida ou não. Embutido em *HTML* e *Javascript*, este campo apareceria preenchido ou não de acordo com certos critérios, anteriormente definidos. Caso fosse uma oferta que já tivesse sido vendida, o *checkbox* deveria aparecer preenchida e desativada, de maneira a que não pudesse ser editada pelos utilizadores. Caso contrário, viria sem preenchimento e com possibilidade de edição. Esta implementação implicou a

comunicação com as entidades da solução de CRM e a verificação dos dados de campos da entidade em questão.

The screenshot shows a table titled 'Oferta' (Offer) with the following columns: Nome da Oferta, Agregado, Macro Família, Família, Sub Família, Código, Produto/Serviço, Família Crédito, Destino Crédito, Linha, Sub Linha, Interesse, Informação, and Venda com Sucesso. The table contains one row with the following data: 209TI201001, Recursos, Passivo, Titulos Investimento, Titulos Investimento, 209TI201001, 209TI201001. The 'Interesse' column has a dropdown menu open with options: Baixo, Médio, and Alto. The 'Venda com Sucesso' column has a checkbox.

Nome da Oferta	Agregado	Macro Família	Família	Sub Família	Código	Produto/Serviço	Família Crédito	Destino Crédito	Linha	Sub Linha	Interesse	Informação	Venda com Sucesso
209TI201001	Recursos	Passivo	Titulos Investimento	Titulos Investimento	209TI201001	209TI201001					Baixo Médio Alto		<input type="checkbox"/>

Figura 3.2 - Criação de um Campo Preenchido Dinamicamente em *Javascript*

Uma outra funcionalidade foi o “*Script Outbound*”. Esta funcionalidade tinha como objetivo auxiliar os operadores do serviço de venda aos consumidores, para os guiarem durante as chamadas telefónicas que tinham com os clientes, com o objetivo de vender as ofertas disponibilizadas pelo banco. A interface disponibilizava a informação que o operador deveria utilizar para comunicar com o cliente e o que responder, conforme o *input* do cliente. Mas a interface permitia também o registo de informação (representado pelos botões da Figura 3.3 “Sem Interesse”, “Reagendar”, “Não Enquadrável”, “Não Atendeu”, “Não quer ser contactado”). Enquanto dialogava com o cliente, o operador tinha a possibilidade de registar o *feedback* do cliente relativamente à oferta que estava a tentar ser vendida pelo operador. As funcionalidades desenvolvidas, no âmbito do estágio, foram a adição do botão “Não quer ser contactado” e a linha abaixo dos botões, que contém a seguinte informação: “Oferta, Interesse, Venda com sucesso”. Novamente, o seu aparecimento era dinâmico pois obedecia a critérios específicos.

Uma outra funcionalidade concretizada foi a possibilidade de navegar livremente entre os vários passos do *script*. Antes, para o operador poder ir ao passo **6. Cópia de Introdução 1** estando no passo **2. Oferta Mulher**, ele teria de percorrer todos os passos intermédios sem poder seleccionar diretamente o passo 6. para o visualizar. A equipa anterior tentou realizar esta funcionalidade, mas sem sucesso. Para cumprir este objetivo, foi necessário modificar um ficheiro de *JQuery* existente que criava estes passos.

Figura 3.3- Desenvolvimentos, em JavaScript, no CRM  
Inclusão da opção “Venda com Sucesso”, do conjunto de opções sobre o “Interesse” e do título da Oferta

Esta fase teve extrema importância para o decorrer do estágio. Foi nesta fase que se aprofundaram os conhecimentos técnicos sobre o CRM, tanto a nível da plataforma em si nativa, como a nível do que poderia ser feito com recursos web. Sem esta experiência, a fase seguinte do estágio (e novo projeto) trariam dificuldades acrescidas.

Em suma, nestes três capítulos foram abordadas as tarefas realizadas em dois projetos:

O **primeiro**, descrito em 3.1.2 - Reestruturação da Solução Atual de CRM: Desenho Funcional, foi desenvolvido para um cliente da **área de serviços energéticos**. Durou somente uma semana, no mês de dezembro, e teve como foco o desenho de processos da solução e o desenvolvimento de documentação funcional. Ou seja, o projeto encontrava-se numa fase de análise e desenho. Tendo em conta que durou apenas uma semana, a maior contribuição incidiu sobre a escrita de alguns capítulos do documento e a reestruturação de alguns desenhos de processos já existentes.

O **segundo**, descrito em 3.1.3 - Implementação de Funcionalidades em *Javascript* e *JQuery*, foi desenvolvido para um cliente do **ramo bancário**. Durou cerca de um mês, em dezembro e janeiro, e incidiu sobre o desenvolvimento de funcionalidades muito específicas sobre a solução atual que o cliente tinha na altura. Ou seja, o projeto encontrava-se numa fase de desenvolvimento. Os

requisitos deste projeto não foram descritos, simplesmente um dos membros da equipa ia pedindo que fossem desenvolvidas certas funcionalidades, à medida que as tarefas iam sendo terminadas.

### **3.2 Desenvolvimentos: Migração de CRM On-Premises para Dynamics 365 na Cloud**

O projeto central deste estágio tem como cliente uma sociedade desportiva (a Organização, referida no início do relatório). Esta Organização possuía uma solução de CRM para gerir os seus fãs e criar oportunidades de negócio. Este modelo foi chamado de B2C – *Business To Consumer*. Antes, utilizavam o Microsoft Dynamics CRM 2015 que integrava com um DW que armazenava os dados dos fãs da Organização. O objetivo do projeto era migrar todos os dados e funcionalidades existentes na solução atual, para uma nova solução de CRM que iria ser criada na nova plataforma da Microsoft, em *cloud*: D365. Esta migração surge porque foram identificados problemas relativamente ao tamanho da base de dados do CRM, que já possuía demasiada informação para o tempo de vida que continha. Este fator aliado a uma implementação com processos pouco eficientes, trouxeram um novo problema: desempenho do sistema. Estas duas situações levaram a que a Organização decidisse que seria altura de modificar o atual paradigma e reestruturar a arquitetura da solução.

Por outro lado, a Organização tinha outro foco: a área *Corporate* ligada ao clube, cujo modelo foi apelidado de B2B – *Business To Business*. Ou seja, todas as entidades parceiras do clube, patrocinadores, entre outros. Esta informação era gerida dentro da mesma solução que geria os Fãs. Para a nova solução desenhada, estes dois módulos – Fãs e *Corporate* – passaram a ser geridos em duas instâncias distintas do D365, de maneira a que a lógica de negócio fosse completamente dividida e que os registos de cada módulo fossem segregados. Esta decisão da equipa baseou-se em dois grandes fatores:

- A entidade nativa do CRM “Contacto” não estava a ser usada de forma apropriada. Estava a representar os contactos associados às empresas que trabalham com a Organização, ao invés de representar um Fã. Esta abordagem torna-se um problema na medida em que a plataforma de CRM foi concebida de forma a que a entidade “Contacto” possa integrar com módulos como o *Customer Care* e o *Social Engagement*, de forma a trocar fluxos de informação com funcionalidades *out-of-the-box*. Para representar um Fã estava a ser



utilizada a entidade “Lead” que foi desenhada para representar uma potencial oportunidade de negócio para a Entidade utilizadora da plataforma do CRM. Esta entidade não contém os atributos nem funcionalidades apropriadas para gerir este tipo de informação, prejudicando todo o fluxo de negócio criado a partir do CRM. Após a identificação destes problemas, foi decidido que iria haver a segregação destes dois negócios, e que a entidade “Contacto” iria ser utilizada para representar o Fã no B2C e, no B2B, seria utilizada para representar os Contactos das Empresas;

- As lógicas de negócio inerentes a estes dois módulos eram (praticamente) independentes uma da outra. Excetuando algumas nuances, não havia informação ou processos que ligassem a camada de negócio do B2C com a de B2B. Sendo assim, era possível implementar processos de sincronização e de integração de forma independente, o que permitiria implementar configurações técnicas específicas a cada instância.

O planeamento do projeto foi desenhado de forma a que, numa primeira fase, fossem desenvolvidas algumas funcionalidades de B2C de forma a poder apresentar ao cliente parte do trabalho. Numa fase mais futura, além da continuidade da implementação do B2C, foram iniciados também os desenvolvimentos relativamente ao B2B.

As próximas subsecções estarão divididas nos dois tipos de negócio mencionados – B2C e B2B – de maneira a abordar as implementações efetuadas nos dois âmbitos, de forma mais organizada e clara.

De notar que o capítulo de B2C conterá mais detalhe sobre o que foi feito do que o capítulo de B2B. Isto porque, a maneira de desenvolver num módulo e noutro foi idêntica, ou seja, as componentes criadas são semelhantes na sua maneira de fazer. Simplesmente muda o contexto e, sendo assim, optou-se por só detalhar num dos capítulos, de modo a não tornar o relatório repetitivo no seu conteúdo.

### **3.2.1 Migração da Solução *Business To Consumer* (B2C)**

Como já foi anteriormente referido, este módulo refere-se a todos os processos de negócio que envolvem os Fãs do clube. Esta instância teve um foco central: conhecer e registar a informação relativamente aos fãs. O objetivo era ter uma **visão de 360°** dos seus fãs, desde os dados pessoais como dados relacionados e até mesmo possíveis ligações entre dois fãs no sistema (um pai e um

filho, por exemplo). Para tal, criou-se uma entidade para armazenar toda a informação sobre os fãs, chamada “Fã” e uma outra entidade chamada “Agregado Familiar” que representava as ligações entre os diferentes fãs.

Noutra perspetiva, a solução desenvolvida permitiu à equipa de **Marketing** da Organização criar campanhas e enviar comunicações baseadas na informação recolhida sobre os seus fãs. Assim, assegurou-se que as suas campanhas teriam uma maior probabilidade em ter sucesso, pois os seus alvos são fãs que cumpriam um determinado critério. Estas funcionalidades foram concretizadas utilizando duas entidades nativas do CRM: Listas de Marketing e Campanhas. Juntas, tornaram possível a criação de listas de fãs que preenchiem determinados critérios (estipulados pelos utilizadores) que associadas a campanhas, criavam oportunidades de negócio à Organização.

Além das capacidades anteriormente referidas, a solução permitia o envio de múltiplos **Alertas de Pagamento** aos fãs. Estes alertas eram gerados no CRM, baseados em informação recolhida por um sistema externo (que foge ao âmbito do relatório). Toda esta componente foi customizada pela equipa, visto serem funcionalidades que não são *out-of-the-box* do CRM. Foram usadas ferramentas disponibilizadas pelo Microsoft Azure – *webjobs* e *Service Bus* – onde os *webjobs* foram agendados para verificarem se estava na altura de enviar alertas de pagamento aos fãs e o *Service Bus* teve o propósito de utilizar as suas filas para o D365 poder comunicar com aplicações externas.

Uma nova componente que foi desenvolvida, relativamente à solução anterior, foi a componente de **Social Engagement** (MSE). É uma plataforma que permite vigiar e interagir com as redes sociais. Ou seja, permite ao utilizador seguir, recolher, analisar e interagir com os clientes (através da rede social). Permite filtrar automaticamente a informação recolhida, eliminando de forma automática informação que seja considerada irrelevante para o contexto [1]. Adicionalmente, esta ferramenta permitiu saber se os sentimentos das conversas dos fãs, nas redes sociais, eram positivos ou negativos e permitiu também identificar publicações que continham determinados *hashtags* ou palavras-chaves. Este módulo pertence ao produto Microsoft D365, mas é um módulo à parte em relação aos restantes descritos.

Outra componente integrada na solução foi a componente de **Customer Care**. Qualquer fã poderá ter uma dúvida, um problema relacionado com um produto que adquiriu ou simplesmente poderá não saber se existe bilhetes à venda para um determinado jogo. Tudo isto tem de ser mitigado e, para tal, utilizou-se um módulo nativo do CRM que permite aos utilizadores registarem incidentes relacionados com os fãs e gerir a resolução dos mesmos. Para fazer esta gestão, foi utilizada uma

entidade nativa do CRM chamada “Incidente”. Esta visão funcional da solução encontra-se resumida na figura abaixo:

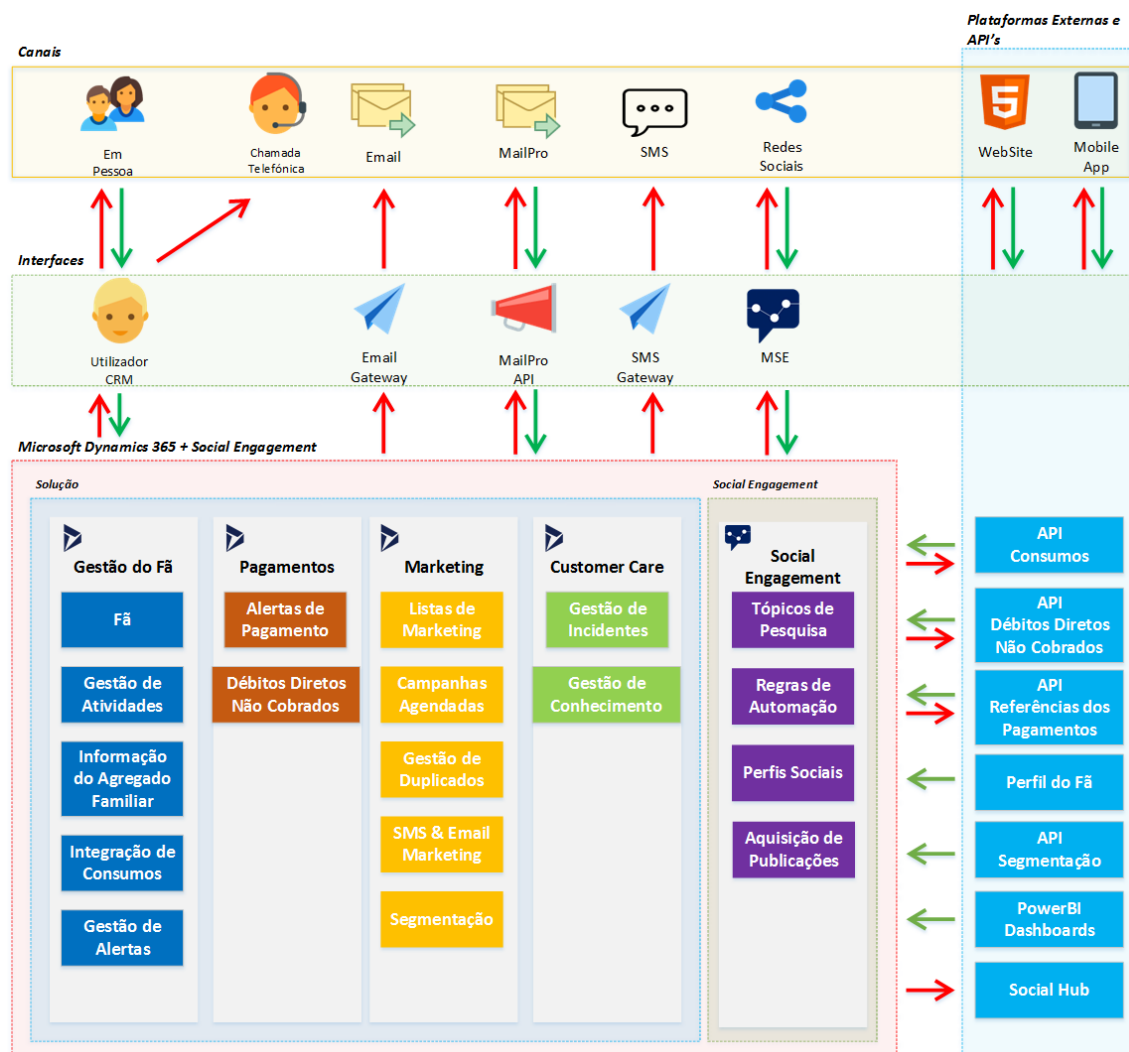


Figura 3.4 - Visão Alto Nível da Solução B2C

### 3.2.1.1 Documentos de Desenho Funcional e Técnico

O projeto foi iniciado na fase de desenho da solução, visto que a equipa já tinha efetuado o levantamento de requisitos. Foram realizados dois documentos essenciais: o documento de desenho funcional e o documento de desenho técnico. Para a realização de ambos os documentos foram necessárias reuniões diárias com o cliente, com o intuito de haver uma validação diária sobre as ideias que estavam a ser criadas no documento. Assim como também foi necessário assistir a

algumas tarefas executadas no CRM atual, pelos utilizadores, de forma a dar uma ideia de clara de todos os processos necessários para conseguir concretizar as funcionalidades que existiam na solução anterior. Paralelamente a isto, foi realizada uma análise às entidades existentes e os seus atributos, de forma a registar essa informação, para na fase de desenvolvimento a poder utilizar. Assim, permitiu-se garantir que a solução nova ficasse coerente com a solução anterior.

## **Documentos de Desenho Funcional**

Este documento pretende descrever ao cliente o comportamento funcional de toda a solução. Desde todo o processo de migração do sistema, os processos que suportam essa mesma migração, mas também as funcionalidades que irão existir em cada componente da nova solução. Serve como ponte para o desenvolvimento do documento técnico e também tem como objetivo ser uma referência para a Organização e, posteriormente, poder produzir um plano de testes.

A Figura 3.4 representa a arquitetura conceptual da solução a ser desenvolvida. A solução conterà quatro camadas: os canais de comunicação entre o Organização e os fãs; as interfaces que possibilitarão aos utilizadores do D365 de realizar as suas tarefas; o D365 em si; as API's disponibilizadas.

As comunicações poderão ser feitas por e-mail, carta, chamada telefónica, redes sociais ou via SMS. Assim, terão de existir vários serviços de integração de modo a que se consigam realizar estas comunicações via CRM (através dos *gateways* para cada canal). A informação proveniente dessas interações com os fãs será devidamente registada na plataforma de modo a que se possa executar ações sobre essa informação, tais como: criação de campanhas e de listas de *marketing*. Esta informação será armazenada no bloco “Gestão de Atividades”. Do lado do Fã, será possível consultar os consumos que cada fã teve. Por consumos, entenda-se, a aquisição de qualquer produto do clube: *merchandising*, bilhética, entre outros.

O documento funcional consistiu na descrição funcional destes módulos tanto a nível dos processos inerentes como das entidades que constituíam cada módulo. Além disto, também conteve a descrição de todo o processo envolvido na migração dos dados.

## **Documento Técnico**

Por sua vez, o documento técnico fornece uma visão técnica da solução ao cliente, especificando os processos e as entidades envolvidas. Este documento é estruturado de uma maneira muito semelhante ao documento funcional, simplesmente cada módulo é abordado de uma visão técnica onde se descreve cada entidade, quais os atributos que vai conter, regras de negócio, fluxos de trabalho, e todas as restantes componentes possíveis.

Neste documento, como já foi mencionado, as entidades envolvidas foram detalhadas ao pormenor. Para tal, foram criados documentos no formato Excel, denominados Desenho da Entidade (*Entity Designs*) - Figura 3.5 - cujo propósito foi elencar cada atributo de cada entidade, com o seu tipo, tamanho, nome, descrição e relação com outro campo de outra entidade (caso fosse o caso). Foram construídos vinte e três documentos desta natureza.

Entity Attributes									
Attributes									
Display Name	Logical Name	Custom	Required	Search	Field Security	Auditing	Type	Size	Description
Detalhes de Pagamento	new_paymentdetailsid	Yes	Required	Yes	No	Yes	Lookup	N/A	Related Payment Details
Nome	new_name	No	Required	Yes	No	Yes	Single Line of Text	160	Payment Alert's name
Agendamento	new_paymentscheduleid	Yes	Required	Yes	No	Yes	Lookup	N/A	Related Payment Schedule
Razão do Estado	statuscode	Yes	No Constraint	Yes	No	Yes	Option Set	N/A	State of the payment alert.
Option Set Values									
Option Set	Value	Description							
Estado	852 790 000	A Processar							
	852 790 001	Enviado							
	852 790 002	Erro no Envio							

Figura 3.5 - Desenho da Entidade “Detalhes de Pagamento”

Foram também desenhados oitenta e quatro *mockups* do que seria a versão final da solução. Na figura abaixo, segue um exemplo de um desses *mockups*, relativamente ao que será um dos formulários que conterá informação sobre os fãs da Organização:

Microsoft Dynamics 365 CRM

http://http:// crm. CRMQA/main.aspx#143754760

+ NOVO CRIAR TICKET ALTERAÇÃO DE DADOS PESSOAIS VER CONSUMOS ...

**Nome do Sócio**

Situação Categoria Cobrador Última Quota Pago

**Dados Pessoais**

Nome

Tipologia

**Outra Informação**

Casa

Sócio Especial

**Cobranças Quotas**

Canal de Cobrança

Morada

Periodicidade

**Tickets**

Título do Ticket	Nº da Ocorrência	Assunto	Tipo de Ticket	Estado
Atualização de Informação	INC-0001-F7R8R0	Bilhética	Venda Ilegal	Ativo
Pedido de Informação	INC-0001-F7R8R1	Bilhética	Bilheteira Estádio	Ativo

**Atividades**

Publicações Atividades Notas

Tudo Adicionar Cham... Adicionar Em... ..

Seguros

Assunto 22/02/2017 16:45

Seguros

Assunto 22/02/2017 16:45

Figura 3.6 - Mockup de Formulário do Fã no Documento Técnico

Após a conclusão de ambos os documentos e validação pelo cliente, segue-se a fase de desenvolvimento da solução em si. O próximo capítulo irá abordar estes desenvolvimentos.

### 3.2.1.2 Desenvolvimentos

Nesta componente de negócio, foram desenvolvidos quatro grandes blocos:

- Customização da entidade Fã
- *Customer Care*
- Aplicação de Negócio
- Gestor de Alertas

De forma a clarificar estas componentes, segue o seguinte esquema ilustrativo:

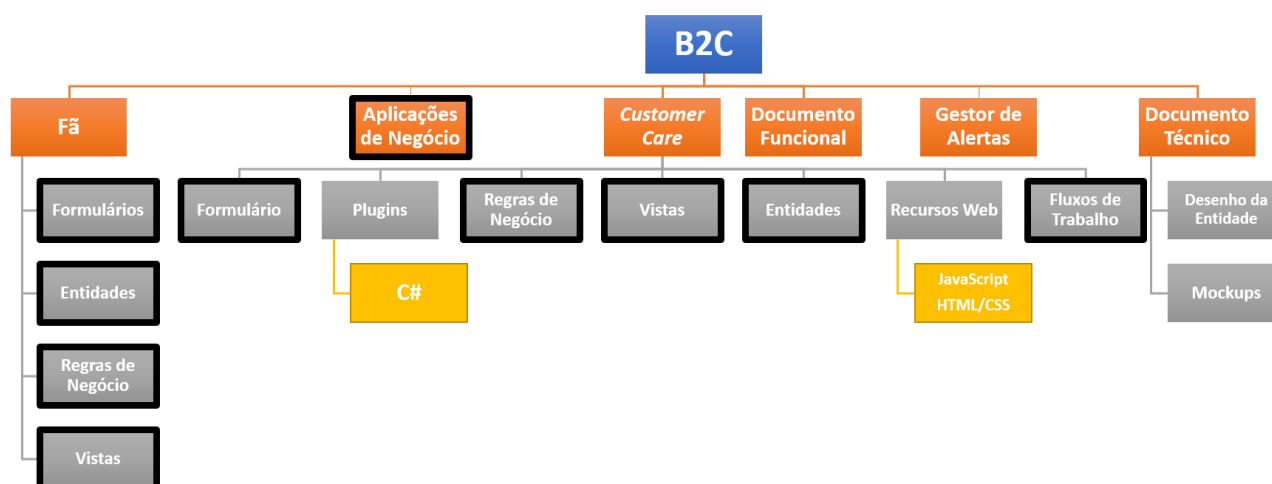


Figura 3.7 - Componentes Desenvolvidas no B2C  
Componentes desenvolvidas encontram-se no segundo nível da hierarquia, agrupadas pelos blocos do sistema (primeiro nível da hierarquia)

No esquema, as diferentes componentes estão agrupadas pelos principais grupos existentes neste projeto: a documentação, o fã, *customer care*, gestor de alertas e aplicações de negócio. Além disto, as componentes que se encontram salientadas (com um traço à volta da caixa mais grosso) representam as componentes nativas do CRM, ou seja, as componentes *out-of-the-box*. Já as que não estão realçadas, são componentes que integram com o CRM, nomeadamente os *Plugins* – desenvolvidos em C# – assim como os recursos web – desenvolvidos em JavaScript ou HTML/CSS. Toda a parte da documentação é um grupo à parte dos restantes.

Cada um destes blocos irá ser detalhado nos tópicos seguintes do presente relatório.

### 3.2.1.2.1 Customização da entidade Fã

Em termos do módulo do Fã, foram desenvolvidas as seguintes componentes: a entidade em si (juntamente com os seus trezentos e onze atributos), os formulários da entidade, regras de negócio e regras de visibilidade de campos. Uma restrição que existe sobre esta entidade é o facto de só ser permitida a criação de Fãs não-sócios (ou seja, adeptos) pois os sócios do clube eram criados no CRM via integração com uma plataforma externa, ficando os registos desses sócios apenas passíveis de leitura por parte dos utilizadores.

Para representar o Fã, foi utilizada a entidade nativa “Contacto” do CRM. Esta entidade já tem os seus atributos nativos criados onde, certa parte, foi utilizada para a caracterização do Fã. Mas, além destes, foi necessário criar outros não-nativos. Após a sua criação, criaram-se dois formulários do

Fã: um para as pessoas que pertenciam ao Departamento de Marketing e um outro para os que pertenciam ao serviço do *Customer Care*. Esta diferenciação deve-se ao facto das pessoas do *Customer Care* só necessitarem de conhecer uma informação parcial do Fã. Desta forma, poupou-se aos utilizadores um formulário muito extenso e com informação irrelevante para o seu trabalho do dia a dia.

## Formulário do Fã

Outra diferenciação que houve foi o formulário para um Fã adepto e para um Fã sócio. Um sócio do clube possui muito mais dados do que um mero adepto, nomeadamente, toda a informação associada às quotas de sócio por exemplo. Contudo, para alcançar este objetivo, não se criaram dois formulários distintos para sócios e adeptos. Havia um só formulário para ambos, mas com regras de negócio e de visibilidade de campos associadas a ele.

Foram desenvolvidas seis regras de negócio, contempladas na seguinte tabela:

#	Nome da regra	Critérios
1	Visibilidade do campo “Categoria Anterior” na criação de um Adepto	SE “Criado em” contém dados <b>Ocultar</b> “Categoria Anterior” <i>CASO CONTRÁRIO</i> Mostra “Categoria Anterior”
2	Mostrar/ocultar os campos referentes aos sócios	SE “Tipologia” = Sócio <b>Mostra</b> campos * <i>CASO CONTRÁRIO</i> Oculta campos *
3	Visibilidade dos campos das Piscinas/Modalidades	SE “Utente Piscinas/Modalidades” = Sim <b>Mostra</b> campos * <i>CASO CONTRÁRIO</i> <b>Ocultar</b> campos *”
4	Visibilidade de campos das quotas modalidades	SE “Quota Modalidades” = Sim <b>Mostra</b> campos * <i>CASO CONTRÁRIO</i> <b>Ocultar</b> campos *”
5	Visualização dos campos “data” e “motivo de demissão”	SE “Situação” = Demitido <b>Mostra</b> “Data de Admissão” && “Motivo da Demissão” <i>CASO CONTRÁRIO</i> <b>Ocultar</b> “Data de Admissão” && “Motivo da Demissão”
6	Visibilidade dos campos das escolas de futebol	SE “Utente Escolas de Futebol” = Sim <b>Mostra</b> campos * <i>CASO CONTRÁRIO</i> <b>Ocultar</b> campos *”

Tabela 1 - Regras de Negócio no Formulário do Fã

**\*Nota:** casos em que foram mostrados/ocultados muitos campos e que ocupariam demasiado espaço na tabela, não sendo relevante a informação de quais os campos que foram ou não ocultados, mas sim a lógica por detrás da regra de negócio.



### **Formulário do *Customer Care* (no âmbito do Fã)**

Tendo em conta que as pessoas do *Customer Care* necessitam de visualizar os dados dos Fãs, foi necessário criar um formulário especificamente para eles (e ao qual só eles tinham visibilidade). Este formulário continha uma quantidade de informação consideravelmente inferior à do formulário do Fã, visto que as pessoas do *Customer Care* não necessitam de toda a informação que reside na entidade do Fã. Sendo assim, criou-se um formulário exclusivamente para o Departamento do *Customer Care*, de forma a fornecer somente os dados necessários para poderem cumprir com o seu trabalho diário. Um exemplo deste tipo de formulário encontra-se em Anexo V.

#### **3.2.1.2.2 *Customer Care***

Como já foi referido este módulo tem como função dar suporte aos utilizadores de forma a que possam ajudar ou prestar serviço aos fãs do clube, desde assistência às suas compras, pagamentos, até à disponibilidade em responder a qualquer tipo de questões colocadas pelo fã. Todos estes “problemas” originam um *Ticket*. Este termo é representado, no CRM, por uma entidade nativa chamada “incidente”. Cada registo guarda informações sobre o Fã que originou o *ticket*, o tipo de problema, a sua resolução, as interações que ocorreram entre o Fã e o utilizador, entre outros.

Para este bloco foram criados os atributos extra necessários da entidade “incidente” para cumprir com os requisitos do cliente. Estes atributos foram os seguintes (nota: o tipo do atributo está entre parenteses):

- Título do *Ticket* (Uma linha de texto)
- Fã ao qual o *Ticket* se destina (Pesquisa)
- Tipo (Conjunto de opções)
  - Sugestão
  - Reclamação
  - Pedido de Informação
  - Operação
  - Venda Ilegal
  - Angariação
- Canal de origem do *Ticket* (Conjunto de opções)
  - Telefone
  - *E-mail*
  - *Web*
  - Facebook / Twitter

- Presencial
- Assunto – baseado na árvore de assuntos do CRM definidas na solução (Pesquisa)
- Descrição (Várias linhas de texto)
- Departamento Responsável pela resolução do *Ticket* (Pesquisa)
- Departamento de Origem: departamento que originou o *Ticket* (Pesquisa)
- Nº de ocorrência: identificador único do *Ticket* (Uma linha de texto)
- Nº de dias de resolução: campo que informava o número de dias que o *Ticket* demorou até ser fechado (Número inteiro)

## Construção do Formulário

Após a criação dos atributos, foi necessário criar o formulário com o qual o utilizador interage, como está ilustrado no Anexo VI.

Para construir este formulário, foi necessário aceder à solução em que ele estava inserido e personalizá-lo. Para tal, é necessário escolher os atributos necessários (secção 1 da Figura 3.8) e dispô-los da maneira correta, através de uma ação de *drag and drop* onde se arrasta o atributo pretendido para a secção 2, representada na Figura 3.8.

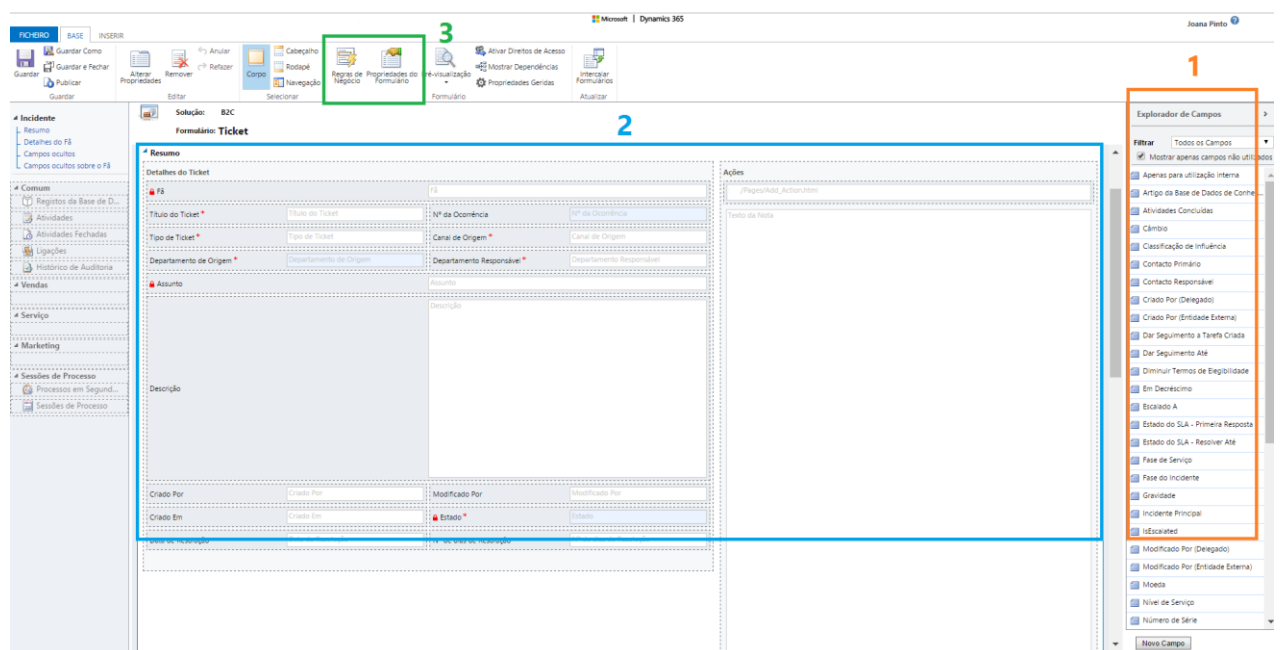


Figura 3.8 - Menu de Personalização do Formulário do *Ticket*

Além das ações de *drag and drop* que são o que, no fundo, permitem “desenhar” o formulário, existem outros dois menus que permitem aplicar regras de negócio (e outros tipos de regras) ao

formulário: “Regras de Negócio” e “Propriedades do Formulário”. Para aceder a estes dois menus, basta clicar nos botões assinalados na secção 3 da Figura 3.8.

Nas “Propriedades do Formulário”, adicionam-se/removem-se bibliotecas de JavaScript para serem usadas nos eventos do formulário e dos atributos. Na secção 1 da Figura 3.9 adicionam-se as bibliotecas (que foram adicionadas à solução de CRM como recurso web).

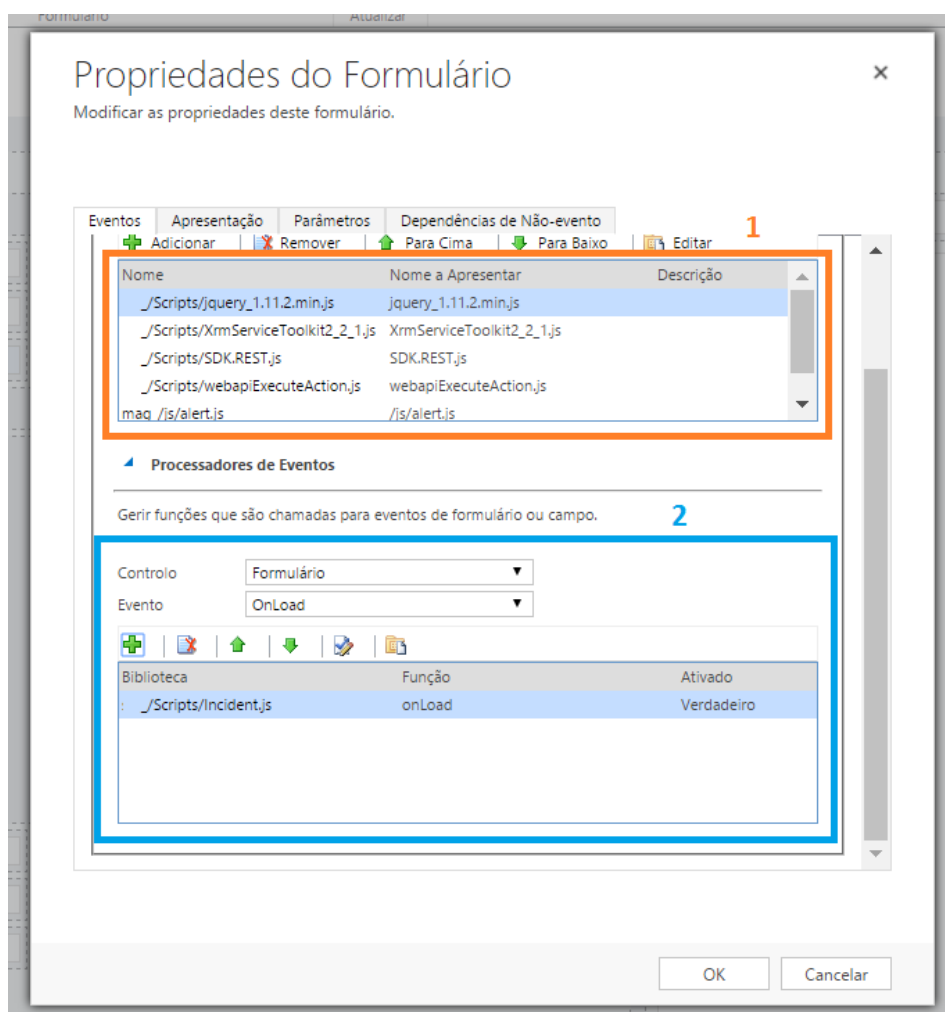


Figura 3.9 – Menu de Propriedades do Formulário  
Interface onde se adicionam recursos web ao formulário, de modo a adicionar lógica adicional ao formulário

Na secção 2 indica-se em que eventos essa biblioteca irá ser executada (Figura 3.11 - *Seleção do Evento da IU (Propriedades do Formulário)*), bem como em que controlo: o formulário, campos ou separadores (Figura 3.10). Neste exemplo, será disparada a função **onLoad()**, a partir do ficheiro de JavaScript chamado **incident.js**, quando o evento *onLoad* do formulário for acionado. Este é um evento que dispara quando a janela de um formulário é aberta no *browser*.

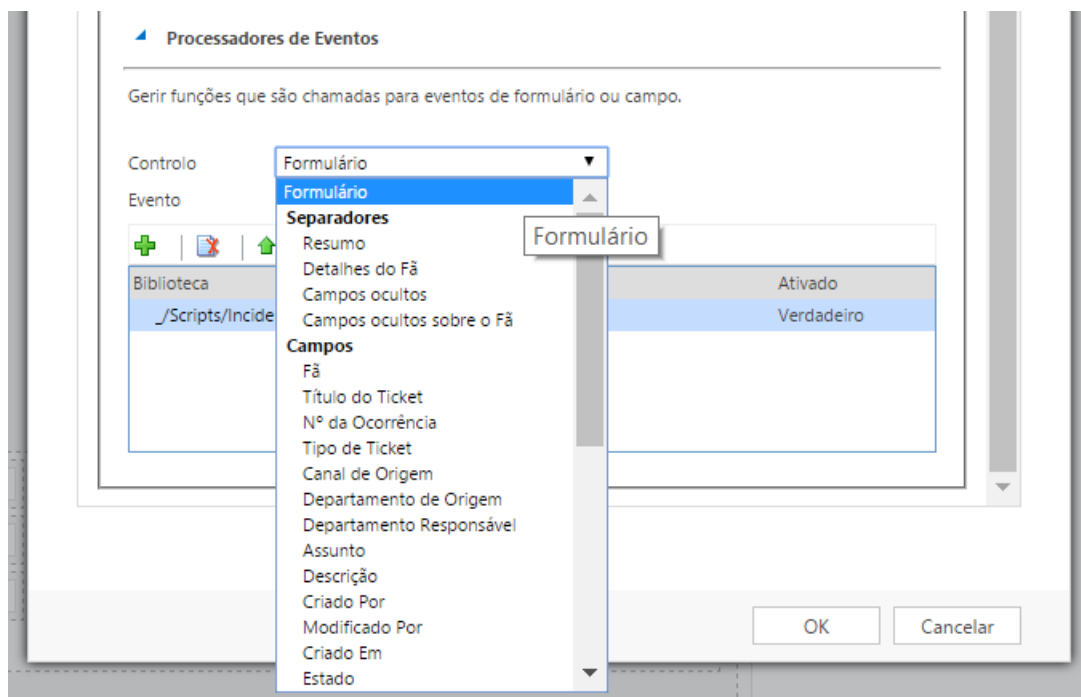


Figura 3.10 - Seleção do Controlo (Propriedades do Formulário)

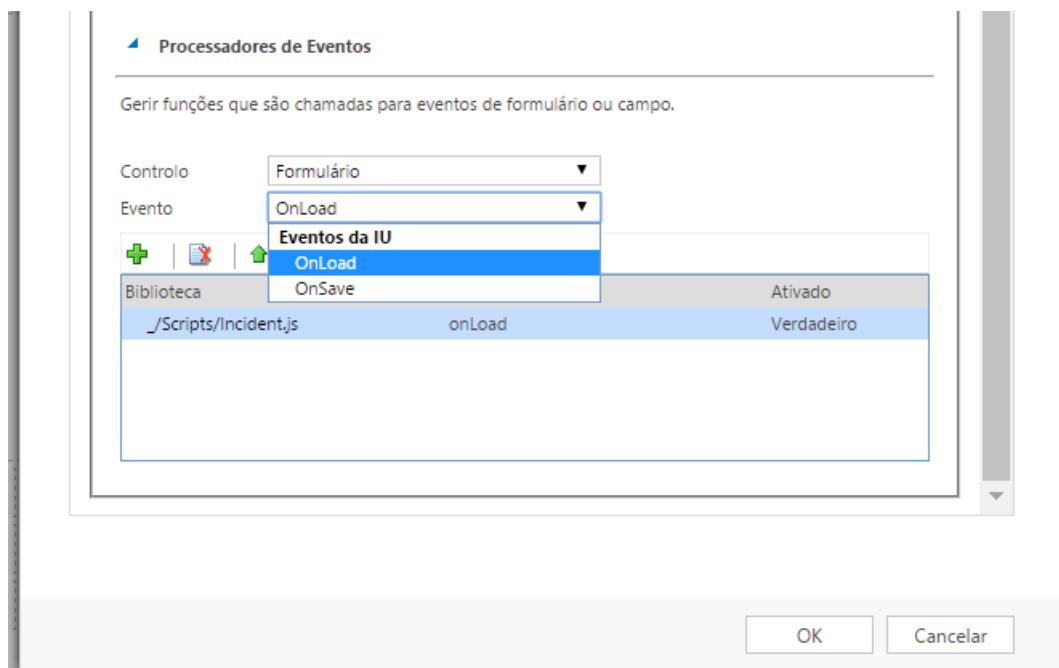


Figura 3.11 - Seleção do Evento da IU (Propriedades do Formulário)

De notar que para diferentes controlos existem diferentes eventos. Para o controlo “Formulário”, como se pode constatar através da Figura 3.11, existem dois eventos possíveis: **onLoad** (evento disparado quando a página do formulário está a ser carregada) e **onSave** (evento disparado quando se gravam as alterações efetuadas no formulário). Contudo, para um controlo “Campo” o evento possível é outro: **onChange**, que é disparado quando é detetada uma alteração no campo escolhido.

## Regras de negócio

O **fluxo do processo de negócio** para a criação de um *ticket* é o seguinte:

1. Identificação do Fã com o problema no sistema (caso o Fã não existisse no sistema, o utilizador criava-o);
2. Criação do *Ticket* e sua categorização;
3. Determinar o Departamento responsável para a sua resolução (ou, se possível, resolver o *Ticket*);
4. Reencaminhar o *Ticket* para a pessoa responsável;
5. Registar as ações que levaram à resolução do *Ticket*;
6. Fecho do *Ticket*.

A este processo estavam associadas e foram desenvolvidas **regras de negócio**, nomeadamente:

1. O Departamento associado ao utilizador que criou o ticket ser automaticamente preenchido no campo “Departamento de Origem” (para não ser necessário o utilizador introduzi-lo manualmente);
2. Inibir o utilizador de criar um *Ticket* caso o utilizador não esteja associado a um Departamento;
3. O campo “Nº de ocorrência” ser automaticamente calculado e preenchido, aquando a criação do *Ticket*;
4. O campo “Nº de dias de resolução” ser automaticamente calculado e preenchido (este requisito foi desenvolvido por um colega da equipa);
5. A atualização da informação do *ticket* só pode ser feita caso o Departamento Responsável do *ticket* seja o mesmo que o do utilizador que está a visualizar o registo e caso o *ticket* ainda não tenha sido resolvido. Contudo, os únicos campos possíveis de alterar são o “Descrição” e o “Departamento Responsável”, de modo a que o utilizador possa reencaminhar o *ticket* para um departamento mais apropriado e que conseguia resolvê-lo;
6. Assim que o utilizador muda o “Departamento Responsável” do ticket, uma notificação via e-mail é enviada para o departamento selecionado;

7. Um utilizador só poderia fechar o *Ticket* caso tivesse alguma interação/ação a ele associada e caso o departamento do utilizador fosse o mesmo que o departamento responsável do *Ticket* (este requisito foi desenvolvido por um colega da equipa).

Para a **concretização** dos **requisitos 1. e 2.** foi desenvolvida uma função que fazia o seguinte:

- Utilizando a linguagem *FetchXML*, executava uma *query* à base de dados da solução, de modo a procurar o departamento associado ao utilizador atual (passando-lhe o *guid* desse utilizador);
- Utilizando uma biblioteca de JavaScript chamada “XrmServiceToolkit” que contém serviços SOAP, enviou-se o *FetchXML* para ser processado;
- Verificou-se os resultados obtidas na resposta; caso existissem resultados (neste caso seria só um, pois um utilizador só podia estar associado a um departamento) então colocava-se o valor do resultado no campo “Departamento de Origem”. Caso contrário, enviava-se um alerta ao utilizador a avisar que ele não tinha um Departamento associado e, conseqüentemente, não poderia criar o *ticket*. Esta função encontra-se representada na Figura 3.13 - *Código JavaScript desenvolvido para o Customer Care*.

A concretização do **requisito 4.** foi obtida, não através de uma função em JavaScript, mas sim pelo tipo de criação do atributo. Ou seja, inicialmente criou-se o atributo “Nº de dias de resolução” com o tipo de dados igual a “Número inteiro” e tipo de campo igual a “Calculado”. Ou seja, o campo vai ser automaticamente calculado, pelo sistema do CRM, de acordo com as condições inseridas. Estas condições são ilustradas na imagem seguinte:

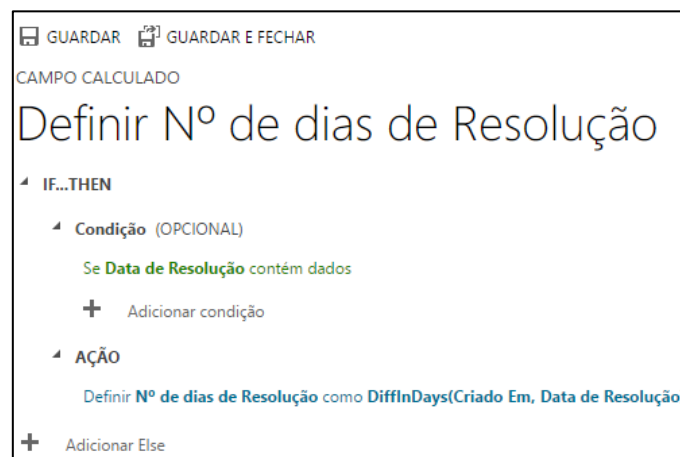


Figura 3.12 - Definição do Campo Calculado “Nº de dias de Resolução”

“DiffInDay(startdate, enddate)” é uma função fornecida pelo CRM que permite calcular a diferença entre dois dias. Esta e outras funções permitem calcular de diversas maneiras campos cujo tipo de dados permita que o campo seja calculado.

```
26 //Altera o campo Departamento de Origem com o Departamento do User
27 function changeOriginDepartment(userId) {
28     var department;
29     var fetchXML = ["<fetch mapping='logical' version='1.0' output-format='xml-platform' distinct='false'>",
30                     "<entity name='systemuser'>",
31                         "<attribute name='fullname' />",
32                         "<attribute name='businessunitid' />",
33                         "<attribute name='departmentid' />",
34                         "<filter type='and'>",
35                             "<condition attribute='systemuserid' operator='eq' value='" + userId + "' />",
36                         "</filter>",
37                     "</entity>",
38                     "</fetch>"];
39     var fetchResults = XrmServiceToolkit.Soap.Fetch(fetchXML.join(""));
40
41     if (fetchResults.length > 0) {
42         department = fetchResults[0].attributes["departmentid"];
43         if (department != null && department != "") {
44             var lookupUser = new Array();
45             lookupUser[0] = new Object();
46             lookupUser[0].id = department.id;
47             lookupUser[0].name = department.name;
48             lookupUser[0].entityType = department.logicalName;
49             Xrm.Page.getAttribute("origindepartmentid").setValue(lookupUser);
50         }
51     }
52     else {
53         Alert.show("Utilizador sem departamento", "O seu Utilizador não tem Departamento configurado. "+
54                 "Não será possível criar um novo Ticket.",
55                 [{
56                     label: "Ok",
57                     callback: function () {
58                     }
59                 }], "ERROR", 500, 200);
60     }
61 }
62
63 }
64 else {
65     Alert.show("Utilizador não existe no sistema", null,
66     [{
67         label: "Ok",
68         callback: function () {
69         }
70     }], "ERROR", 500, 200);
71 }
72 }
73 }
```

Figura 3.13 - Código JavaScript desenvolvido para o Customer Care

O requisito 5. divide-se em duas partes:

- uma função JavaScript para concretizar a parte “A atualização da informação do ticket só pode ser feita caso o Departamento Responsável do ticket seja o mesmo que o do utilizador que está a visualizar o registo e caso o ticket ainda não tenha sido resolvido”;
- Uma regra de negócio out-of-the-box do CRM para a segunda parte “Os únicos campos possíveis de alterar são o “Descrição” e o “Departamento Responsável” de modo a que o utilizador possa reencaminhar o ticket para um departamento mais apropriado e que conseguia resolvê-lo”.

A regra de negócio era muito simples: a condição era validar se o *Ticket* já tinha sido criado. Para tal, bastou verificar se o atributo “Criado Em” continha dados (visto que este atributo é automaticamente preenchido pelo sistema assim que um registo é criado). No caso afirmativo, bloqueava os campos desejados. Caso contrário, desbloqueava. Esta regra de negócio pode ser visualizada no Anexo V. Além da definição das possíveis condições (caixas com linha a tracejado) e ações (caixas com linha normal), é também necessário definir em que âmbito a regra de negócio vai ser disparada. Este âmbito refere-se aos formulários onde ela irá ser disparada (ilustrado no canto superior direito do Anexo V). Neste caso, esta regra fazia sentido ser disparada em qualquer formulário do *Ticket*.

Por fim, para o **requisito 6.**, foi necessário construir um Fluxo de Trabalho de forma a cumprir com o requisito. Antes de definir o que o Fluxo de Trabalho deve fazer, é necessário definir:

- Como deve ser executado o fluxo: em segundo plano, a pedido ou como um processo subordinado;
- O âmbito: organização, utilizador, unidade de negócio ou nível principal: unidades de negócio subordinadas;
- Quando deve ser iniciada a sua execução: aquando a criação do registo, da mudança do estado do registo ou de um atributo, entre outros.

Todas estas definições são feitas no menu ilustrado no Anexo VI. Para este caso em concreto, o Fluxo de Trabalho foi criado de forma a que fosse executado após a alteração do atributo “Departamento Responsável” do *Ticket*. A alteração deste campo significava que o *Ticket* tinha sido atribuído a algum novo Departamento e, consequentemente, teria de ser criada uma notificação a avisar esse Departamento.

Seguiram-se todas as validações e ações do fluxo em si. Fizeram-se duas validações: se o campo “Departamento Responsável” continha dados onde, no caso negativo, o fluxo terminava; se o utilizador que fez o fluxo de trabalho disparar era um utilizador de integração o fluxo também terminaria. E executou-se duas ações: obter o nome da fila para a qual seria enviada a notificação e a obtenção da fila em si, seguida da criação do email para o departamento. Todo este processo está demonstrado na figura seguinte:



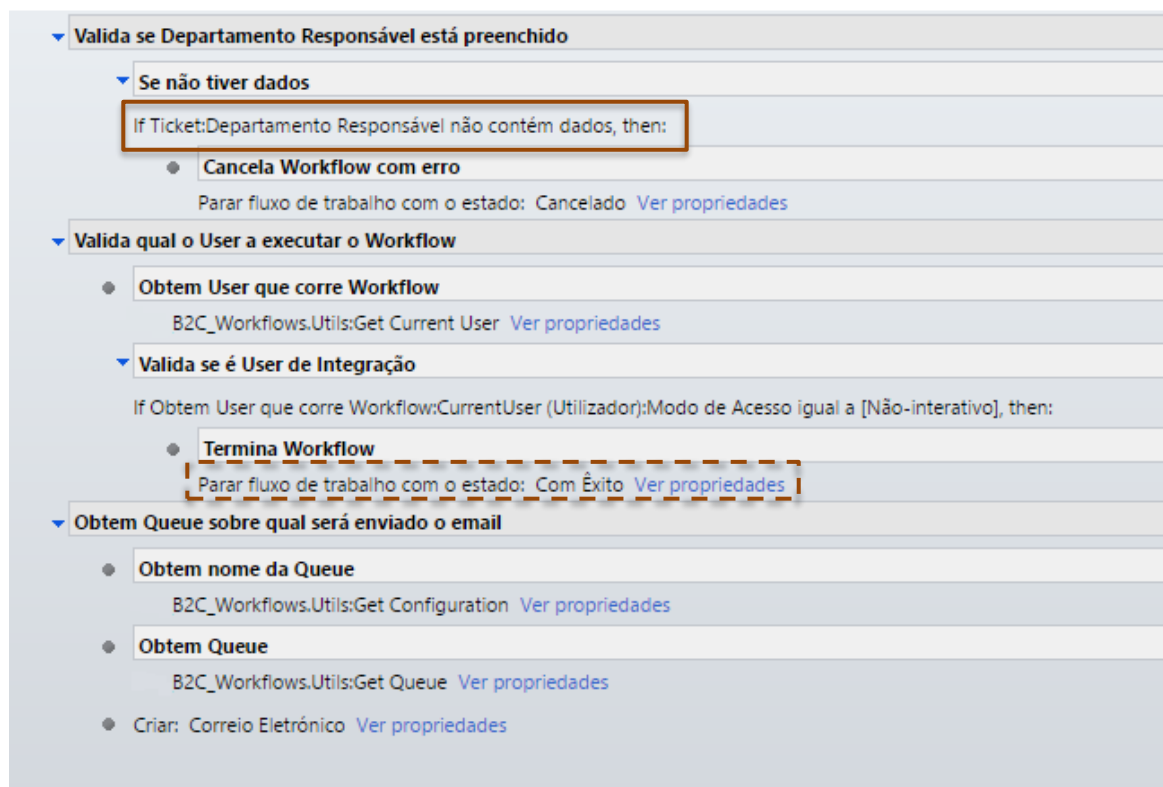


Figura 3.14 - Construção do Fluxo de Trabalho  
Validações (traço normal) e ações (tracejado)

## Pedido de alteração de dados de um Sócio

De forma a manter os dados pessoais de um Sócio coerentes com a realidade, é necessário mantê-los atualizados no sistema de CRM. Para tal acontecer, esses dados têm de ser facultados pelo Sócio. Uma das formas dele informar o clube que houve alguma alteração nos seus dados pessoais (por exemplo, uma mudança no seu e-mail) era comunicar com o serviço de *Customer Care* e informá-los dessa alteração. Contudo, um funcionário do *Customer Care* não podia alterar diretamente, no CRM, os dados desse Sócio. Para esta alteração ser consumada, o funcionário teria de criar um *Ticket* especial e reencaminhá-lo para o departamento responsável apropriado para estes poderem validar e efetuar a alteração. Assim garantia-se uma uniformidade da informação e centralização da mesma (visto que só este Departamento é que tinha autorização para alterar dados de um Sócio).

Para este desenvolvimento, foi adicionado um novo botão na *ribbon* do formulário do Sócio (*Customer Care*) designado “Alteração de Dados Pessoais”. Ao clicar neste botão, o utilizador seria reencaminhado para o formulário de um novo registo de uma nova entidade customizada chamada “Alteração de Dados Pessoais (*contactinformationupdate*)”. Esta entidade foi criada com o

propósito de auxiliar esta operação. Os seus atributos eram idênticos aos atributos dispostos no formulário de um Fã-Sócio, permitindo que o utilizador modificasse os atributos desejados e permitidos. Assim que o *Ticket* fosse criado com a informação dos atributos a modificar, o registo era automaticamente eliminado.

O fluxo deste processo é o seguinte:

- Clicar no botão “Alteração de Dados Pessoais”;
- Criação automática de um registo da entidade “*contactinformationupdate*” com os atributos pré-preenchidos (com os valores que estavam no registo do Sócio) e possíveis de editar;
- Alteração, por parte do utilizador, dos campos desejados;
- Clicar no botão de “Guardar”;
- Criação automática de um registo de um *Ticket*, com os seguintes campos pré-preenchidos:

Nome do Atributo	Valor do Atributo
Título do Ticket	Alteração de dados de <Nome do Sócio>
Tipo do Ticket	Operação
Assunto	Alteração de dados pessoais
Descrição	Sumário dos pedidos de alteração. Tinha a seguinte sintaxe: <Atributo>: “Valor do atributo antigo” → “Valor do atributo novo”

Tabela 2 - Campos Pré-Preenchidos do Ticket (Alteração dos Dados Pessoais)

- Escolha do Departamento Responsável;
- Gravar o *Ticket*.

Para **concretizar** esta funcionalidade, foram desenvolvidas três funções em JavaScript:

- *updateInformationCheckType()*: função que mostra o botão “Alteração de Dados Pessoais” caso o Fã fosse do tipo “Sócio”; ocultava o botão caso contrário;
- *openAndFillContactUpdate()*: função disparada no momento em que o botão “Alteração de Dados Pessoais” é clicado. Nesta função, o formulário da entidade de suporte “*contactinformationupdate*” é aberto para o utilizador e pré-preenchido com os mesmos atributos do Sócio-alvo;
- *preFillTicket()*: função disparada no evento **onSave** do formulário do Ticket, com a informação alterada na entidade de suporte. Este método valida quais os campos que tinham sido modificados e compila-os na descrição do Ticket.

A seção do *Customer Care* foi a seção com mais desenvolvimentos, visto que foi também a área em que o cliente tinha os seus requisitos muito bem definidos, o que permitiu à equipa de desenvolvimento uma maior produtividade neste âmbito.

### 3.2.1.2.3 Aplicações de Negócio

Aplicações de Negócios (em inglês, *Business Apps*) são, no fundo, um agrupamento das entidades relacionadas, *dashboards* e fluxos de processo de negócio que permitem ao utilizador navegar pela solução de CRM em diferentes áreas de negócio, podendo, no entanto, só visualizar as componentes que lhe são permitidas – estas permissões são atribuídas através dos *roles* de segurança dos utilizadores.

Para a solução desenvolvida foram criadas duas Aplicação de Negócios: **Marketing** e **Customer Care**. A primeira destina-se aos utilizadores da solução que pertencem ao Departamento de Marketing do clube. Já a segunda, destina-se às pessoas do Departamento de *Customer Care*. Estas aplicações encontram-se visíveis para o utilizador através da barra lateral do D365, como a figura abaixo demonstra:

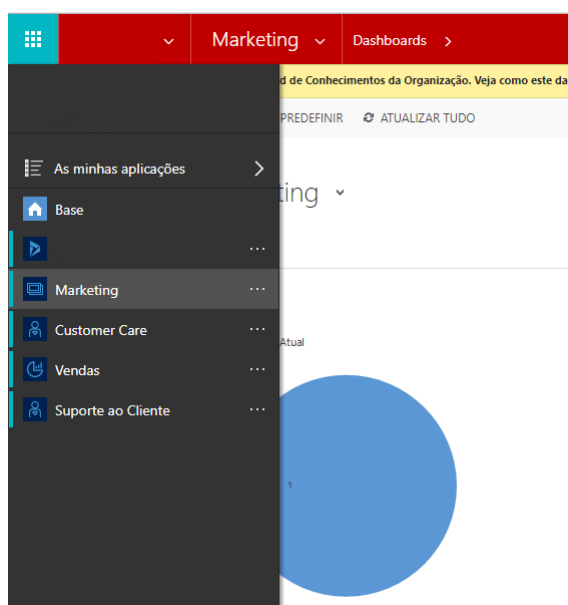


Figura 3.15 - Acesso às Aplicação de Negócios

Para configurar estas aplicações, utiliza-se uma funcionalidade própria do D365. Como a Figura 3.16 ilustra, é necessário definir alguns aspetos da aplicação, nomeadamente:

- Nome a ser mostrado ao utilizador da aplicação;

- Nome lógico;
- As entidades que vão pertencer à aplicação;
- Os formulários e vistas visíveis aos utilizadores;
- O ícone da aplicação;
- Os *dashboards* e gráficos;
- Mapa do site (Figura 3.17).

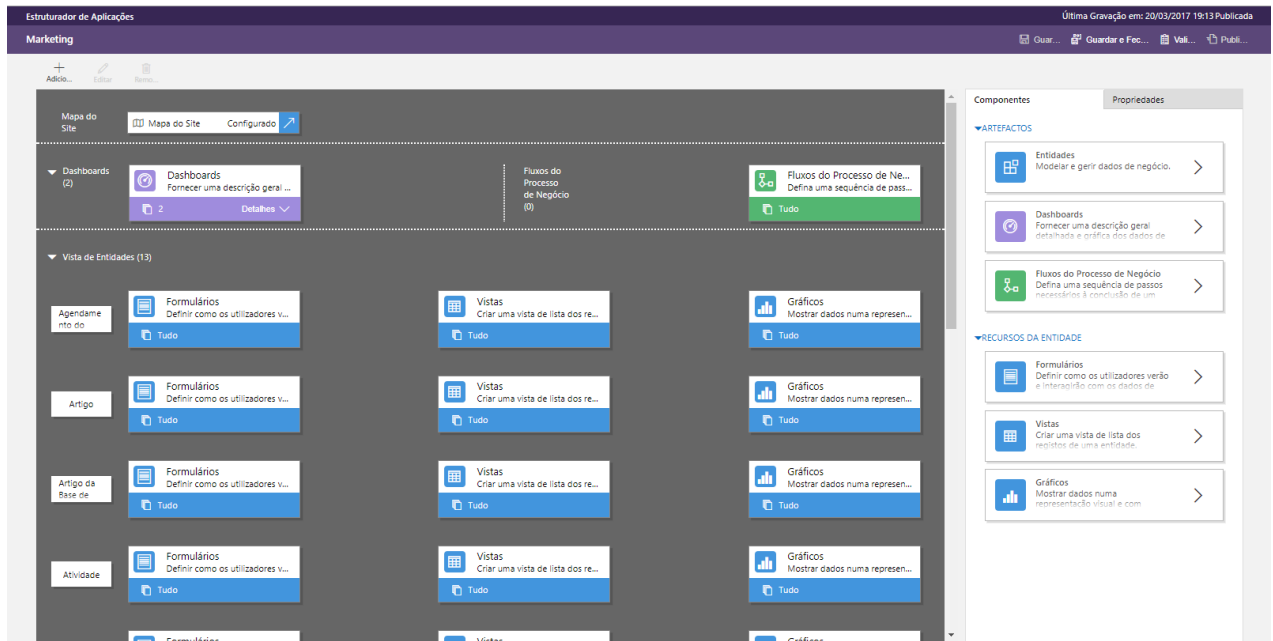


Figura 3.16 - Estruturador de Aplicações de Negócio

Para desenhar o mapa do site do módulo do Marketing, é necessário definir as suas componentes. Na lista seguinte, cada componente estará representada na Figura 3.17 abaixo com a respetiva caixa indicada:

- **Áreas** (caixa com linha tracejada pequena), onde se define o nome, o ID e o ícone;
- **Grupos** (caixa com linha normal), onde se define o título e o ID;
- **Subáreas** (caixa com linha tracejada grande), onde se define o tipo de componente (entidade, *dashboard*, recurso web ou URL), a componente em si, o título, o ícone e o ID.

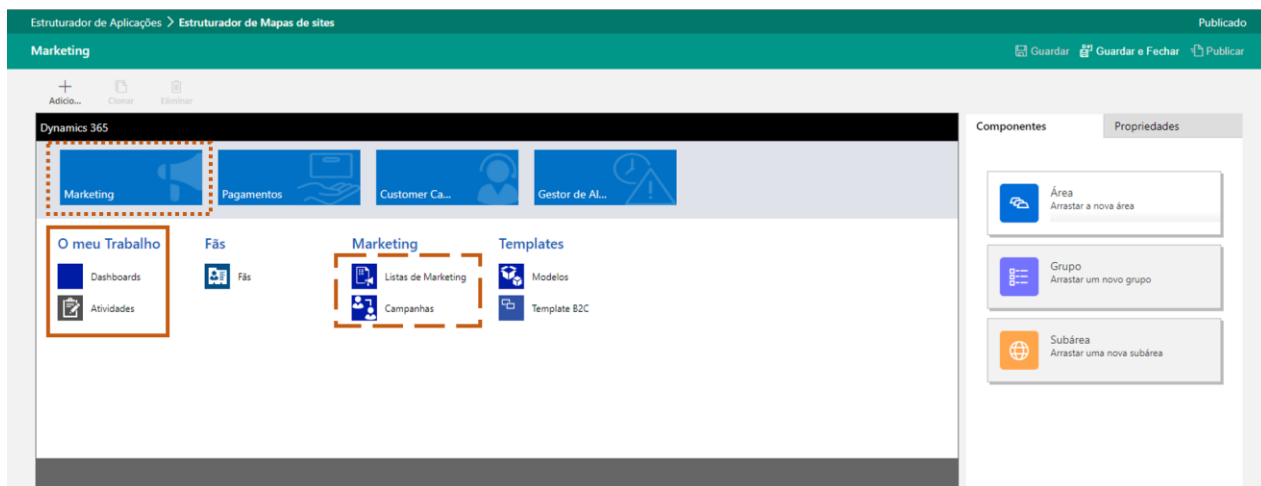


Figura 3.17 - Mapa do Site do Módulo de Marketing

## Marketing

Os utilizadores com acesso a esta aplicação, tinham acesso às seguintes componentes e funcionalidades da solução:

- Entidade do Fã;
- Formulário do Fã (todos menos o do *Customer Care*, referido no Anexo III);
- Entidades das Campanhas e Listas de Marketing;
- Gestor de Alertas (que será detalhado em 3.4.1.2.4);
- Alertas de Pagamentos.

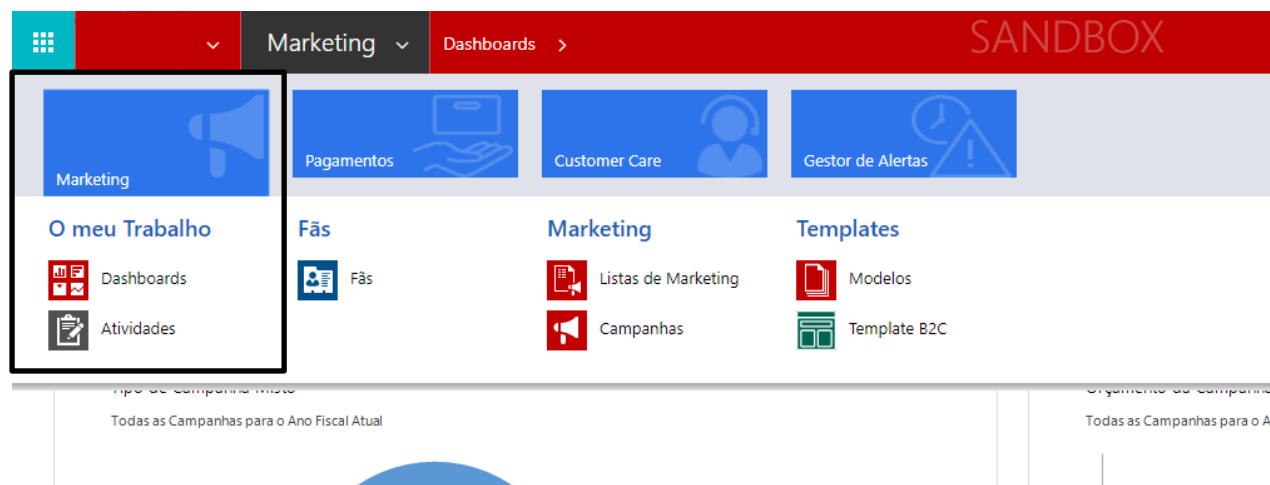


Figura 3.18 - Aplicação de Negócio de Marketing

Para cada bloco selecionado (neste caso, o bloco de Marketing), o utilizador consegue visualizar diferentes entidades (que estão relacionadas com esse bloco).

### ***Customer Care***

O Departamento de *Customer Care* tem acesso às seguintes componentes e funcionalidades:

- Entidade do *Ticket*;
- Entidade do Fã;
- Formulário do Fã (*Customer Care*);
- Formulário do *Ticket*;
- Todas as funcionalidades desenvolvidas, previamente descritas, relativamente ao reencaminhamento do *ticket*, criação de um *ticket* para alterar os dados pessoais de um sócio, entre outros.

Estas aplicações permitem que, para uma dada solução, a organização que utiliza a solução de CRM consiga separar de forma clara e simples os seus departamentos internos e que não haja conflito entre o que cada departamento pode visualizar e/ou editar dentro da solução.

#### **3.4.1.2.4 Gestor de Alertas**

O gestor de alertas refere-se a uma funcionalidade, pedida pela Organização, que alertava os utilizadores do departamento de Marketing, aquando a abertura de um registo de uma entidade previamente definida, sobre determinadas informações relevantes para o âmbito de negócio da organização. Por exemplo, se um utilizador entrasse no registo de uma Fã chamada “Joana Pinto” e se essa fã tivesse cotas em atraso, então uma notificação era disparada e mostrada ao utilizador (no topo do formulário dessa fã) de forma a alertar a organização sobre esse dado em falta e relevante.

Para tornar esta funcionalidade possível nesta solução, foi criada uma entidade customizada chamada “Gestor de Alertas” que continha os seguintes atributos:

- Nome do alerta (Uma linha de texto)
- Mensagem que era disparada no alerta (Uma linha de texto)
- Prioridade (Conjunto de opções) – de 1 a 5

- Tipo (Conjunto de opções)
  - Informação
  - Aviso
  - Erro
- Entidade sobre a qual o alerta iria ser disparado (Conjunto de opções - recurso web)
- Vista da entidade sobre a qual o alerta iria ser disparado (Conjunto de opções - recurso web)

A **prioridade** existia para o caso de existirem mais do que um alerta e poder ordenar a sua apresentação no formulário, de forma ascendente. Por exemplo, se existissem dois alertas X e Y sendo que o X tinha uma prioridade com o valor 1 (a mais alta) e o Y com o valor 5, no formulário seria apresentado em primeiro lugar a mensagem do alerta X e só depois o Y.

O **tipo** de alerta servia para ser apresentado um determinado ícone associado ao tipo escolhido. Esta visualização servia somente para o utilizador entender logo o tipo de alerta que foi disparado, de forma a que a mensagem complementasse a mensagem do alerta.

A entidade e vista da entidade foram desenvolvidas num ficheiro de HTML que, no fundo, apresentava ao utilizador uma *dropdown* semelhante à existente no D365. Não podendo utilizar o conjunto de opções disponibilizado pelo D365, houve a necessidade de se criar esta funcionalidade como um recurso web visto que era necessário executar uma pesquisa sobre o sistema CRM, de forma a obter as entidades disponíveis bem como as vistas da entidade escolhida (e isto não é possível alcançar através das funcionalidades *out-of-the-box* do D365). Para tal, foram construídos os seguintes métodos em JavaScript:

- *getViews(element)*: método que ia buscar as vistas de uma dada entidade;
- *fillDropdown()*: preenchia a *dropdown* com as vistas disponíveis, de forma a que o utilizador pudesse seleccionar a desejada;

As restantes funcionalidades necessárias para a concretização desta funcionalidade foram desenvolvidas por outro colega da equipa, não fazendo parte do âmbito deste relatório.

Como se pode verificar, os desenvolvimentos previamente descritos abrangeram um grande leque de funcionalidades do CRM: desde as funcionalidades *out-of-the-box* como as regras de negócio e os fluxos de trabalho, até ao desenvolvimento de recursos web em JavaScript. Tudo isto enriqueceu

os conhecimentos técnicos do CRM, tendo sido este o período mais enriquecedor e produtivo de todo o estágio.

### 3.4.2 Migração da Solução de *Business To Business* (B2B)

A componente de B2B focou-se na gestão das empresas que trabalham em conjunto com a Organização. O núcleo desta implementação residiu no conhecimento e informação relativamente às Empresas, o que permitia à Organização ter uma visão de 360º graus sobre as mesmas.

Tal como sucedeu no contexto do negócio em B2C, também nesta componente foram desenvolvidas funcionalidades de forma a que o Departamento Comercial e de Marketing (designado **DCM**) pudesse criar campanhas para um alvo específico. Para este departamento foi também criado um módulo de criação e gestão de **Propostas** – que representam uma oferta feita pela Organização à empresa cliente em troca de alguma compensação financeira. Por fim, para o DCM, foi criado um módulo de criação e gestão de **Eventos** Comerciais. A estes eventos estavam associados dois tipos de informação: o local onde o evento foi organizado e os seus participantes. Porém, nem todas as pessoas podiam ser convidadas para os eventos, pelo que tinham num campo a informação sobre se estavam ou não inibidos de serem convidados para eventos.

Existiram ainda dois blocos adicionais e independentes do DCM: **Fundação** da Organização, que armazena informação relacionada com os projetos e ações sociais do clube, e as **Casas**. O módulo das casas foi desenvolvido de forma a guardar toda a sua informação, num único repositório, servindo só para processos de criação e consulta de informação. Esta arquitetura é semelhante à arquitetura da solução de B2C: existem três camadas, sendo elas os canais, as interfaces e a plataforma do D365 (sendo esta alimentada por várias API's disponibilizadas por entidades externas). Abaixo segue uma imagem que representa a arquitetura conceptual da solução B2B, desenvolvida e descrita neste relatório:



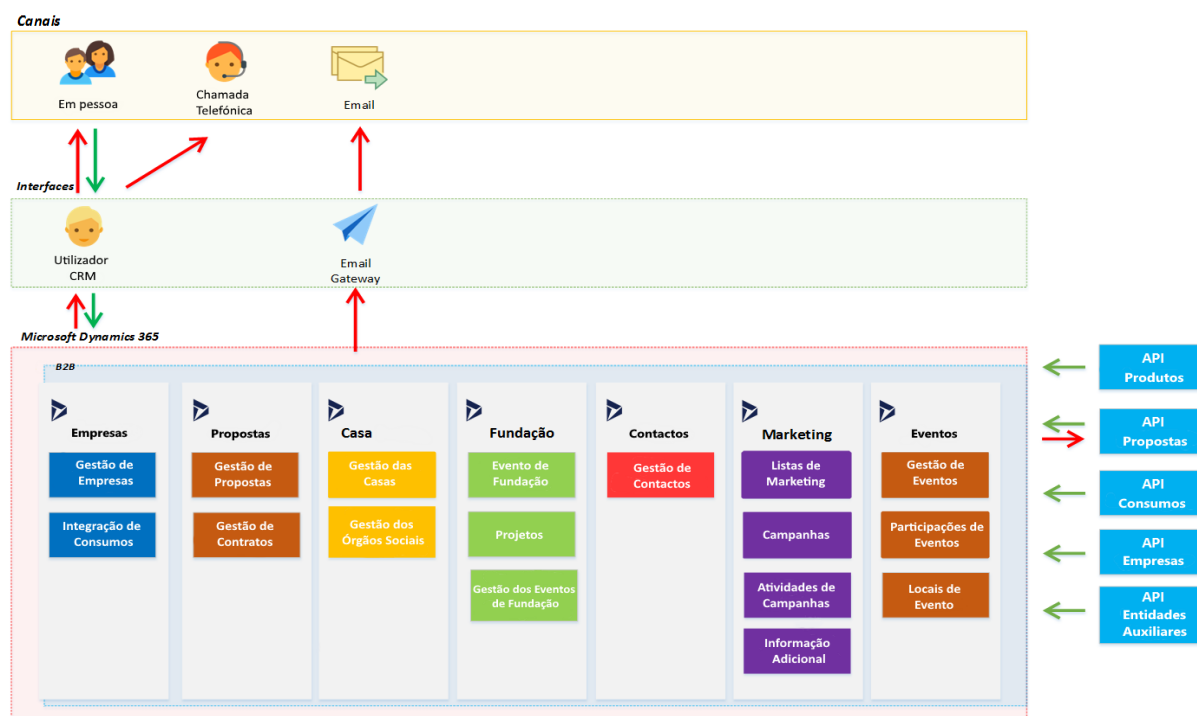


Figura 3.19 - Visão de Alto Nível da solução B2B

### 3.4.2.1 Documento de Desenho Funcional e Técnico

De forma análoga ao que sucedeu na camada de negócio do B2C, foi necessário desenvolver dois documentos: o documento de desenho funcional e o técnico. O primeiro focou-se na descrição, a nível funcional, das funcionalidades a serem desenvolvidas e no desenho dos processos de negócio a elas associadas. O segundo tinha como objetivo descrever tecnicamente como seria desenvolvida a solução, nomeadamente, descrever como seriam constituídas as entidades que iriam ser criadas (através dos, já referidos, Desenhos da Entidade). Aqui foram apresentados trinta e nove *mockups* das interfaces futuras da solução, assim como trinta e um documentos de Desenho da Entidade.

### 3.4.2.2 Desenvolvimentos

Nesta componente de negócio foram desenvolvidas customizações (e por customizações entenda-se criação/edição de entidades, formulários, regras de negócio, recursos web, aplicações de negócio, entre outros) na solução do CRM, ao nível dos seguintes módulos já previamente referenciados:

- Empresas
- Propostas

- Eventos
- Fundação
- Casas

Foram também desenvolvidas **aplicações de negócio**, à semelhança do que foi feito para o módulo de B2C. Assim sendo, foram desenvolvidas quatro aplicações de negócio: Comercial, Fundação, Eventos e Marketing.

Numa fase final da participação neste projeto, foi também realizada a migração de alguns dados da Organização.

À semelhança da subsecção dos desenvolvimentos em B2C, segue a seguinte figura que ilustra, de forma geral e compacta, os desenvolvimentos efetuados para o negócio do B2B:

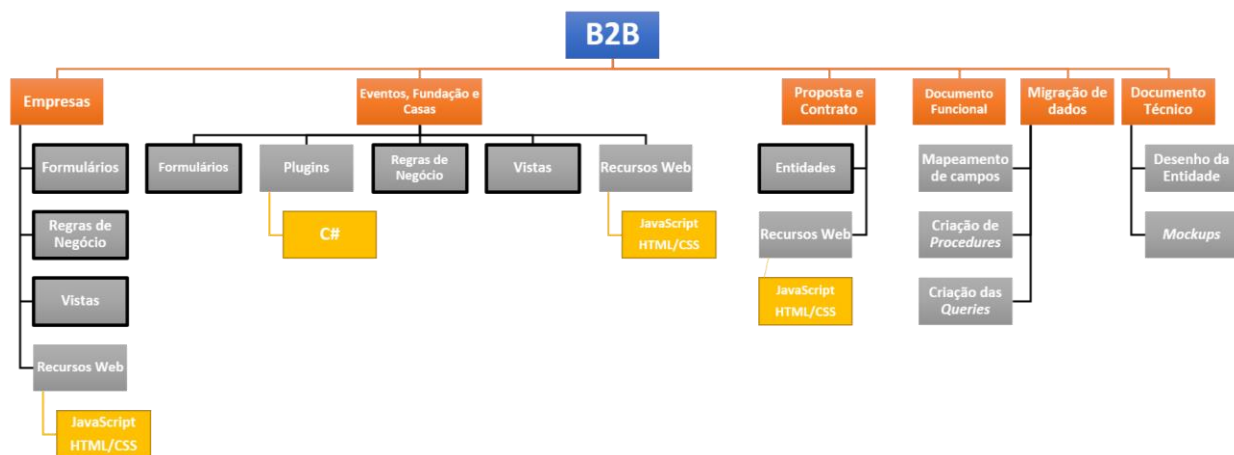


Figura 3.20 - Componentes Desenvolvidas no B2B

Componentes desenvolvidas encontram-se no segundo nível da hierarquia, agrupadas pelos blocos do sistema (primeiro nível da hierarquia)

Para o negócio do B2B, o fluxo de trabalho foi muito similar ao do B2C. A grande diferença reside na migração de dados, que foi realizada para B2B mas não para B2C. Excetuando este caso, todo o trabalho incidiu na criação de entidades, os seus formulários, regras de negócio, vistas e, se necessário, lógica adicional através de *plugins* e recursos web.

Os próximos tópicos irão abordar cada um destes temas, de forma mais aprofundada, mas menos detalhada que na secção 3.2.1.2 visto que a lógica por detrás destes desenvolvimentos é muito

semelhante aos descritos na secção referenciada o que levaria a uma repetição no conteúdo do relatório.

#### **3.4.2.2.1 Empresas e Propostas**

Para representar e guardar os dados das Empresas, foi usada a entidade nativa do CRM denominada “conta” (em inglês, *account*). Foi criado um formulário para esta entidade que continha várias regras de visibilidade de campos associadas a ele. Isto deveu-se ao facto de existir uma tipologia para as empresas. Cada empresa podia ter várias tipologias associadas a ela, em simultâneo, exceto para alguns casos. Para cada tipo, existia um conjunto específico de atributos. Esses atributos eram apresentados (ou não) no formulário, conforme as tipologias associadas à empresa.

Um dos tipos possíveis era a empresa representar uma “Casa” do clube, como as filiais. Quando assim o era, existiam órgãos sociais relacionadas com essa Casa. Quando um utilizador seleccionava este tipo para uma empresa, tinha também de seleccionar a que Casa a empresa se referia (visto que existia um módulo independente que guardava as informações relativamente às Casas). Para esta tipologia, essa informação era relevante e era mostrada no formulário, como o Anexo VII o demonstra.

Contudo, para qualquer outra tipologia, esta informação torna-se irrelevante pois para cada tipologia existe um conjunto de informação específica a essa tipologia. Sendo assim, caso o utilizador definisse um outro tipo para a empresa, essa informação tinha de ser ocultada. O Anexo VII (‘ultima imagem) ilustra um exemplo do que foi desenvolvido para este cenário.

Para o bloco das Propostas, foram também usadas entidades nativas da plataforma de CRM: proposta (*quote*) e contrato (*contract*).

Para ambos os módulos, foi desenvolvido um formulário para cada entidade, as suas vistas, bem como regras de negócio e funções de *JavaScript*.

#### **3.4.2.2.2 Eventos, Fundação e Casas**

Para o caso do módulo dos **Eventos**, foram criadas três entidades: a entidade “Evento” em si, com trinta e um atributos, uma regra de negócio (que impedia que o utilizador inserisse uma data de fim do evento superior à data de início) e um formulário; a entidade “Espaço Evento” que representava

o local onde o evento era executado, que continha vinte atributos e um formulário; a entidade “Participação do Evento” que guardava a informação dos participantes do evento e continha vinte e quatro atributos e um formulário.

### JavaScripts para manipulação de lógica

Para a entidade “Evento”, foi desenvolvido um ficheiro de JavaScript de forma a cumprir com os requisitos dados pelo cliente, para gerir os eventos na solução de CRM:

- *showAvailableEventLocation()*: função que apenas permitia, ao utilizador, que este escolhesse um local disponível (esta disponibilidade obedecia a certas regras) para organizar o evento;
- *getDate()*: função que convertia datas para um formato que fosse compatível com o CRM.

Para a entidade “Participação do Evento” também foi desenvolvido um ficheiro de JavaScript que continha as seguintes funções:

- *checkContactInhibition()*: método que verificava se a pessoa escolhida para aquele evento não estava inibida de ser convidada para eventos;
- *retrieveContactCallBack()*: caso se verificasse que a pessoa não podia ser selecionada como convidado para o evento, era enviada uma notificação para o utilizador de forma a informá-lo sobre este cenário. Esta função era uma função de *callback* que era chamada dentro da função *checkContactInhibition()*.

No módulo da **Fundação**, foram criadas várias entidades, mais precisamente oito entidades. Estas entidades geriam informação sobre os projetos de ação social, os beneficiários destes projetos, os voluntários, os eventos que patrocinavam estas ações, entre outros. Para cada uma delas, foi criado um formulário. Apenas numa destas entidades, foram criadas regras de negócio. As restantes foram entidades cujo propósito era somente guardar e mostrar informação ao utilizador.

Seguindo esta lógica, o módulo das **Casas** serviu também só para guardar a informação das casas, a sua localização, o número de membros de cada uma delas, os órgãos sociais que as compunham, entre outros tipo de informação. Foram criadas duas entidades para este módulo: a “Casa” em si e “Órgão Social”, que continha três relações para a Casa (uma relação para cada tipo de Órgão Social possível – Assembleia Geral, Direção e Conselho Fiscal). Para a entidade “Casa” foi criado um

formulário com várias regras de negócio. Já para o “Órgão Social” foram criados três formulários: um para cada tipo de órgão social possível.

### 3.4.2.2.3 Migração de Dados

Tanto para a solução de B2C como para a B2B, foi necessário migrar os dados que estavam contidos na solução anterior *On-Premises* do CRM da Organização para as duas novas soluções desenvolvidas. A migração dos dados para a componente B2C foi executada por um colega de equipa, ao passo que para a componente B2B a responsabilidade foi individual, através de processos de ETL.

Para executar esta tarefa, foram usados pacotes de *SQL Server Integration Services* <sup>13</sup>(normalmente chamados de SSIS), usando a ferramenta *SQL Server 2014 Management Studio*, que estava provisionada numa máquina virtual no Azure (disponibilizado pelo cliente). Foi também necessário criar um utilizador de integração que tinha permissões para aceder à base de dados da Organização, ao Azure e à solução de D365 (o destino da migração dos dados).

Foi desenhado um plano (pelo colega da equipa especialista em SQL) com passos a executar, de forma a concretização com sucesso a migração dos dados. A tabela seguinte ilustra cada passo desse plano. A tabela é constituída pelas seguintes colunas:

- **#** - ordem de execução dos passos;
- **Passo** – descrição do passo;
- **Ferramentas utilizadas** – nome das ferramentas utilizadas para executar os passos;
- **Figura** – referência à figura, caso exista, que representa o passo descrito (as imagens encontram-se no fim desta tabela).

#	Passo	Ferramentas Utilizadas	Figura
1	Extrair o modelo de dados da solução antiga de CRM.	XrmToolBox (descrita em 2.3.10)	Figura 3.21

---

<sup>13</sup> *SQL Server Integration Services* (SSIS) é uma plataforma que permite construir soluções de transformações e integração de dados, a nível empresarial [19].

2	<p>Extrair o modelo de dados da solução atual.</p>	XrmToolBox	N/A
3	<p>Criar um ficheiro Excel com o mapeamento entre o modelo de dados da solução antiga com o modelo de dados da solução nova. O documento foi dividido em três partes (que correspondiam às três diferentes bases de dados em que os dados se encontravam) – CRM <i>On-Premises</i> (solução antiga), <i>Staging</i> (base de dados intermediária, usada simplesmente para processos de migração) e D365 (solução nova).</p> <p>Em cada uma destas secções no documento, indicavam-se três tipos de informação diferentes: o nome lógico, de apresentação do campo e o seu tipo. Na secção da base de dados do CRM On-Premises indicava-se também se era um campo que era para migrar para a base de dados <i>staging</i> e, consequentemente, para a do D365 da solução atual. Esta decisão era tomada de acordo com a existência ou não de dados no atributo.</p> <p><b>Nota:</b> a coluna “<i>Logical Name</i>” da secção CRM On-Premises mantinha-se sempre visível, de forma a conseguir sempre entender a qual campo original correspondia o novo campo.</p>	Excel	<p>Figura 3.22</p> <p>Figura 3.23</p>
4	<p>Foi criado um documento Excel chamado “<i>Query Template</i>”. Este documento tinha como objetivo criar de forma imediata as <i>queries</i> a serem utilizadas no <i>procedure</i> desenvolvido para migrar a entidade. Aqui contemplou-se somente os atributos que iam ser migrados, ou seja, os atributos filtrados pelo campo “<i>Field to Migrate</i>” no documento referido no ponto anterior. Este documento era constituído pelo nome antigo do atributo, pelo seu novo nome, pelo tipo de dados e depois, pelas <i>queries</i> que representavam o seguinte:</p> <ul style="list-style-type: none"> <li>a. <i>Generate Field: query</i> que criava os campos na BD de <i>staging</i>;</li> <li>b. <i>Generate Query SELECT: query</i> que selecionava os atributos a migrar da BD de <i>staging</i> para a de desenvolvimento (ou seja, na BD para onde iriam os dados migrados);</li> <li>c. <i>Generate Query UPDATE: query</i> usada para atualizar os dados na BD de desenvolvimento, para os casos de atualização e não de criação dos dados;</li> <li>d. <i>Generate Query CREATE: query</i> que criava os campos na BD de desenvolvimento;</li> <li>e. <i>Generate Query SELECT (INSERT): query</i> que inseria os dados na BD de desenvolvimento.</li> </ul> <p>Estas <i>queries</i> eram executadas no SQL Server 2014 Management Studio.</p>	Excel; SSMS	<p>Figura 3.24</p> <p>Figura 3.25</p>
5	<p>Após colocar as <i>queries</i> criadas no <i>procedure</i>, foi necessário executar esse <i>procedure</i> de forma a que criasse as bases de dados com a informação migrada.</p>	SSMS	N/A

6	Por último, foram criados pacotes de SSIS que se ligavam a esta base de dados criada e migravam os dados da BD para a solução de D365 desenvolvida.	Visual Studio 2014; KingswaySoft	N/A
---	---	--	-----

Tabela 3 - Passos a Executar no Processo de Migração de Dados

Logical Name	Display Name	Type	Additional data
createdby	Criado Por	Lookup	Targets:systemuser
createdbyname	N/A	String	Format: TextMax length: 100
createdbyyominame	N/A	String	Format: TextMax length: 100
createdon	Criado Em	DateTime	Format: DateAndTime
createdonbehalfby	Criado Por (Delegado)	Lookup	Targets:systemuser
createdonbehalfbyname	N/A	String	Format: TextMax length: 100
createdonbehalfbyyominame	N/A	String	Format: TextMax length: 100
importsequencenumber	Número de Sequência da Importação	Integer	Minimum value: -2147483648Maximum value: 2147483647
modifiedby	Modificado Por	Lookup	Targets:systemuser
modifiedbyname	N/A	String	Format: TextMax length: 100
modifiedbyyominame	N/A	String	Format: TextMax length: 100
modifiedon	Modificado Em	DateTime	Format: DateAndTime
modifiedonbehalfby	Modificado Por (Delegado)	Lookup	Targets:systemuser
modifiedonbehalfbyname	N/A	String	Format: TextMax length: 100
modifiedonbehalfbyyominam	N/A	String	Format: TextMax length: 100
organizationid	ID da Organização	Lookup	Targets:organization
organizationidname	N/A	String	Format: TextMax length: 100
overriddencreatedon	Registo Criado Em	DateTime	Format: DateOnly
new_channelid	Canal	Uniqueidentifier	
new_channelname	Nome do Canal	String	Format: TextMax length: 100
new_mdmid	Código	String	Format: TextMax length: 100
statecode	Estado	State	States:0: Ativo1: Inativo
statecodename	N/A	Virtual	
statuscode	Razão do Estado	Status	States:1: Ativo2: Inativo
statuscodename	N/A	Virtual	
timezoneruleversionnumber	Número de Versão da Regra de Fuso Horário	Integer	Minimum value: -1Maximum value: 2147483647

Figura 3.21 - Extração do Modelo de Dados da Entidade “Canal” do CRM

CRM On-perm					
Logical Name	Display Name	Type	Additional data	Field to Migrate	
createdby	Criado Por	uniqueidentifier	Targets:systemuser	YES	
createdbyname	N/A	String	Format: TextMax length: 100	NO	
createdbyyominame	N/A	String	Format: TextMax length: 100	NO	
createdon	Criado Em	DateTime	Format: DateAndTime	NO	
createdonbehalfby	Criado Por (Delegado)	uniqueidentifier	Targets:systemuser	NO	
createdonbehalfbyname	N/A	String	Format: TextMax length: 100	NO	
createdonbehalfbyyominame	N/A	String	Format: TextMax length: 100	NO	
emailaddress	Endereço de Correio Eletrônico	nvarchar(100)	Format: EmailMax length: 100	YES	
importsequencenumber	Número de Sequência da Importação	int	Minimum value: -2147483648Maximum value: 2147483647	NO	
modifiedby	Modificado Por	uniqueidentifier	Targets:systemuser	NO	
modifiedbyname	N/A	String	Format: TextMax length: 100	NO	
modifiedbyyominame	N/A	String	Format: TextMax length: 100	NO	
modifiedon	Modificado Em	DateTime	Format: DateAndTime	NO	
modifiedonbehalfby	Modificado Por (Delegado)	uniqueidentifier	Targets:systemuser	NO	
modifiedonbehalfbyname	N/A	String	Format: TextMax length: 100	NO	
modifiedonbehalfbyyominame	N/A	String	Format: TextMax length: 100	NO	
organizationid	ID da Organização	uniqueidentifier	Targets:organization	YES	
organizationidname	N/A	String	Format: TextMax length: 100	NO	
overriddencreatedon	Registro Criado Em	DateTime	Format: DateOnly	NO	
new_name	Nome do Vendedor	String	Format: TextMax length: 100	YES	
new_vendorid	Vendedor	Uniqueidentifier		YES	
new_vendorinitial	Inicial Comercial	String	Format: TextMax length: 10	YES	
statecode	Estado	int	States:0: Ativo1: Inativo	YES	
statecodename	N/A	Virtual		NO	
statuscode	Razão do Estado	int	Status:1: Ativo2: Inativo	YES	
statuscodename	N/A	Virtual		NO	
timezoneruleversionnumber	Número de Versão da Regra de Fuso Horário	Integer	Minimum value: -1Maximum value: 2147483647	NO	
utcconversiontimezonecode	Código de Fuso Horário de Conversão UTC	Integer	Minimum value: -1Maximum value: 2147483647	NO	
versionnumber	N/A	BigInt	Minimum value: -9223372036854775808Maximum value: 9223372036854775807	NO	

Figura 3.22 - Mapeamento do Modelo de Dados entre Solução Antiga e Nova  
(Dados da Base de Dados Antiga)

Staging Database				D365		
Field Name	Table Name	Type	Additional Mapping	Logical Name	Display Name	Type
createdby	new_vendor	uniqueidentifier	Targets:systemuser	createdby	Criado Por	uniqueidentifier
createdbyname	new_vendor	String	Format: TextMax length: 100	createdbyname	N/A	String
createdbyyominame	new_vendor	String	Format: TextMax length: 100	createdbyyominame	N/A	String
createdon	new_vendor	DateTime	Format: DateAndTime	createdon	Criado Em	DateTime
createdonbehalfby	new_vendor	Lookup	Targets:systemuser	createdonbehalfby	Criado Por (Delegado)	Lookup
createdonbehalfbyname	new_vendor	String	Format: TextMax length: 100	createdonbehalfbyname	N/A	String
createdonbehalfbyyominame	new_vendor	String	Format: TextMax length: 100	createdonbehalfbyyominame	N/A	String
emailaddress	new_vendor	nvarchar(100)		emailaddress	Correio Eletrônico	nvarchar(100)
importsequencenumber	new_vendor	Integer	Minimum value: -2147483648Maximum value: 2147483647	importsequencenumber	Número de Sequência da Importação	Integer
modifiedby	new_vendor	Lookup	Targets:systemuser	modifiedby	Modificado Por	Lookup
modifiedbyname	new_vendor	String	Format: TextMax length: 100	modifiedbyname	N/A	String
modifiedbyyominame	new_vendor	String	Format: TextMax length: 100	modifiedbyyominame	N/A	String
modifiedon	new_vendor	DateTime	Format: DateAndTime	modifiedon	Modificado Em	DateTime
modifiedonbehalfby	new_vendor	Lookup	Targets:systemuser	modifiedonbehalfby	Modificado Por (Delegado)	Lookup
modifiedonbehalfbyname	new_vendor	String	Format: TextMax length: 100	modifiedonbehalfbyname	N/A	String
modifiedonbehalfbyyominame	new_vendor	String	Format: TextMax length: 100	modifiedonbehalfbyyominame	N/A	String
organizationid	new_vendor	uniqueidentifier	Targets:organization	organizationid	ID da Organização	uniqueidentifier
organizationidname	new_vendor	String	Format: TextMax length: 100	organizationidname	N/A	String
overriddencreatedon	new_vendor	DateTime	Format: DateOnly	overriddencreatedon	Registro Criado Em	DateTime
new_name	new_vendor	nvarchar(100)	Format: TextMax length: 100	new_name	Nome	nvarchar(100)
new_vendorid	new_vendor	uniqueidentifier		new_vendorid	Vendedor	uniqueidentifier
new_vendorinitial	new_vendor	nvarchar(10)	Format: TextMax length: 10	new_vendorinitial	Inicial Comercial	nvarchar(10)
statecode	new_vendor	int	ints:0: Ativo1: Inativo	statecode	Estado	int
statecodename	new_vendor	Virtual		statecodename	N/A	Virtual
statuscode	new_vendor	int	ints:1: Ativo2: Inativo	statuscode	Razão do Estado	int
statuscodename	new_vendor	Virtual		statuscodename	N/A	Virtual
timezoneruleversionnumber	new_vendor	Integer	Minimum value: -1Maximum value: 2147483647	timezoneruleversionnumber	Número de Versão da Regra de Fuso Horário	Integer
utcconversiontimezonecode	new_vendor	Integer	Minimum value: -1Maximum value: 2147483647	utcconversiontimezonecode	Código de Fuso Horário de Conversão UTC	Integer
versionnumber	new_vendor	BigInt	Minimum value: -9223372036854775808Maximum value: 9223372036854775807	versionnumber	N/A	BigInt

Figura 3.23 - Mapeamento do Modelo de Dados entre Solução Antiga e Nova  
(Dados da Base de Dados de Staging e da Nova)



OLD Fieldname	NEW/Staging Fieldname	DataType	Generate Field
createdby	createdby	uniqueidentifier	[createdby] uniqueidentifier NULL,
createdon	createdon	DateTime	[createdon] DateTime NULL,
new_quotetypeid	new_quotetypeid	Uniqueidentifier	[new_quotetypeid] Uniqueidentifier NULL,
new_quotetypename	new_name	nvarchar(44)	[new_name] nvarchar(44) NULL,
statecode	statecode	int	[statecode] int NULL,
statuscode	statuscode	int	[statuscode] int NULL,

Figura 3.24 – Documento de Apoio à Migração da Entidade "Proposta"

Generate Query SELECT	Generate Query Update	Generate Query CREATE	Generate Query SELECT (INSERT)
,crm.createdby	,d365.createdby = crm.createdby	,createdby	,crm.createdby
,crm.createdon	,d365.createdon = crm.createdon	,createdon	,crm.createdon
,crm.new_quotetypeid	,d365.new_quotetypeid = crm.new_quotetypeid	,new_quotetypeid	,crm.new_quotetypeid
,crm.new_quotetypename AS new_name	,d365.new_name = crm.new_name	,new_name	,crm.new_name
,crm.statecode	,d365.statecode = crm.statecode	,statecode	,crm.statecode
,crm.statuscode	,d365.statuscode = crm.statuscode	,statuscode	,crm.statuscode

Figura 3.25 - Documento de Criação de *Queries* para Migração da Entidade "Proposta"

Após a execução destes pacotes (que davam de imediato *feedback* sobre o sucesso, ou não, da operação), verificava-se se todos os atributos tinham sido devidamente migrados. Foram migradas trinta e duas entidades, onde cada uma delas passou pelo processo explicado na Tabela 3 - *Passos a Executar no Processo de Migração de Dados*, tendo sido este o último desenvolvimento feito para este projeto.

Estes desenvolvimentos foram extremamente enriquecedores pois incidiram sobre uma área que não foi explorada ao longo do estágio (exceto nesta fase), o que permitiu ganhar experiência numa área diferente, aumentando assim o leque de conhecimentos adquiridos no estágio.



## Capítulo 4.

### Conclusão

De seguida, descrevem-se as principais contribuições do presente Projeto de Engenharia Informática (PEI), as competências adquiridas no decorrer do estágio, as principais dificuldades encontradas bem como possíveis desenvolvimentos futuros.

#### 4.1 Principais Contribuições

Tal como foi referido no início do presente relatório, foram delineados objetivos a serem cumpridos até ao término do estágio. Estas contribuições incidiram sobre os três projetos em que houve envolvimento.

Sendo assim, as principais contribuições para o projeto principal – onde o cliente era uma organização desportiva - incidiram sobre o negócio do **B2C**, mais especificamente, sobre o desenvolvimento do bloco do *Customer Care*, onde foi garantido um bom desempenho das operações executadas pelos utilizadores. Houve um cuidado extra em, não só garantir um bom desempenho das operações, mas também de apresentar ao cliente uma interface “limpa” e clara aos olhos do utilizador. Para tal, foram realizadas diversas reuniões com a equipa de suporte ao cliente, de forma a analisar corretamente as melhores abordagens para atingir este fim. Houve inclusivamente uma demonstração (individual) do sistema aos utilizadores, de forma a garantir que os desenvolvimentos efetuados cumpriam todos os requisitos. O *feedback* do cliente foi positivo relativamente a esta componente, mostrando-se satisfeito com a fluidez das operações e com o facto de a linha de aprendizagem ao novo sistema não ser demasiado grande. No bloco da gestão do Fã houve igualmente uma contribuição para o projeto, visto que foram criadas todas as entidades (sendo que o Fã era a entidade com mais atributos) e toda a lógica envolvida nos formulários - Figura 3.7.

Outra grande contribuição foi na parte funcional do projeto, onde houve uma grande dedicação no desenvolvimento dos documentos técnicos e funcionais, de forma a estar o maior conteúdo possível documentado – ajudando assim, tanto o cliente, como a equipa. Foi necessária uma grande

componente de análise ao sistema atual e às necessidades expressas pelo cliente, de forma a poder mapear as duas numa nova solução onde o fluxo de trabalho, para os utilizadores, não diferisse muito do que eles tinham atualmente, mas que diferisse o necessário para cumprir com os requisitos estipulados.

Outra grande contribuição foi todo o desenvolvimento e customização da solução de **B2B** que foi feito, na sua maioria, individualmente. Foram criadas, praticamente, todas as entidades e todas as personalizações necessárias, que cumpriam com os requisitos do cliente. O documento funcional foi igualmente realizado individualmente, com a supervisão de um sénior da equipa - Figura 3.20.

No primeiro projeto do estágio, onde o cliente pertencia à área de serviços energéticos, apesar da participação ter sido curta comparativamente com os restantes projetos, houve uma contribuição analítica e funcional para com o projeto. Foi necessário analisar o comportamento da solução anterior e perceber como se poderia integrar com a nova reformulação do sistema. Foram desenhados vários processos que foram, posteriormente, integrados na documentação funcional do projeto.

No cliente da área bancária, a contribuição foi mais a nível técnico do que funcional. A análise funcional do projeto foi feita juntamente com um colega sénior da equipa, onde depois foram listados requisitos específicos a serem cumpridos. Todos eles foram concretizados e realizados com sucesso e, hoje em dia, esses desenvolvimentos figuram na solução do cliente, que se encontra em produção.

Em suma, houve contribuições em todas as vertentes dos projetos, com uma maior incidência sobre o bloco de *Customer Care* do módulo B2C e sobre a solução toda do módulo B2B.

## 4.2 Competências Adquiridas

Este PEI permitiu a utilização de novas e variadas tecnologias, que possibilitaram a concretização dos principais objetivos definidos.

Relativamente ao CRM, foram adquiridas competências na montagem e customização de uma solução de CRM, desde a criação e utilização de entidade, personalização de fluxos de trabalho (e de como eles afetam a solução de CRM), criação de recursos web que escondem e/ou mostram informação (e, consequentemente, foram melhoradas técnicas sobre a linguagem JavaScript).

Apesar do CRM ser um assunto que não foi tratado na Faculdade, houveram alguns aspectos e conceitos que fizeram ponte com o que foi aprendido na Faculdade, nomeadamente:

- Linguagem **JavaScript** – todos os recursos web eram desenvolvidos em JavaScript ou HTML/CSS;
- Noção de **processos síncronos** e **assíncronos** – quando se definia os fluxos de trabalho, era necessário ter noção do que era mais apropriado: executá-los em segundo plano (assíncrono) ou em tempo-real (síncrono);
- Conceitos de **base dados relacionais**: entidades, relações, atributos, entre outros – como já foi mencionado no relatório, uma entidade e tudo o que a envolve no CRM é muito semelhante aos conceitos aprendidos sobre base de dados relacionais;
- Noções sobre **Interface de Utilizador** – na criação dos formulários foi necessário ter em consideração a experiência do utilizador, tentando tornar toda a interação com os formulários o mais *user-friendly* possível.

Foram também adquiridas competências no que ao negócio diz respeito. Muitas vezes, foi necessário pensar e montar estratégias de implementação muito específicas ao negócio em questão. O que implicou sair da zona do conforto e pensar mais além.

A nível social, foram desenvolvidas algumas competências. Desde logo, através das reuniões com o cliente, o que exigiu uma postura e discurso diferentes dos habituais. Também através da comunicação entre a equipa, visto que este era um ambiente empresarial (o que é muito diferente do ambiente académico). A interação com outras equipas do mesmo projeto também ajudou ao desenvolvimento das competências sociais. Todas estas competências adquiridas garantem uma melhor preparação para o futuro: quando o contexto for um emprego real e não um estágio, haverá esta experiência passada que colmatará certas falhas naturais de quem está a iniciar a sua carreira profissional e ajudará a ter uma integração mais rápida e positiva tanto na empresa como nos desafios dos projetos.

A um nível geral, este estágio possibilitou a aquisição de um conjunto de competências específicas do trabalho em empresa, nomeadamente o funcionamento numa equipa mais sénior e a organização de tarefas entre cada membro da equipa. Este aspeto foi muito positivo, visto que forneceu uma visão do mundo empresarial e do trabalho em equipa muito particular, bem como a integração numa equipa multidisciplinar de pessoas com diferentes níveis de experiência e conhecimentos.

### 4.3 Principais Dificuldades

Logo de início, a maior dificuldade foi encontrar a motivação certa para começar o estágio. Isto porque, inicialmente, o PEI incidia sobre a área mobile. Porém, o projeto passou a ser de uma outra área – CRM. É uma área que não se aborda na faculdade e, como tal, foi tudo novo (desde os conceitos até às tecnologias utilizadas). Encontrar a motivação para “mergulhar” num projeto que não era o desejado e onde não havia uma ideia clara do que se tratava, foi desafiante. Porém, estas dificuldades tornaram toda a experiência mais gratificante. Foi necessário estudar um pouco sobre o tema, ser proativo e interrogar os membros das equipas de forma a conseguir superar o desconhecimento do tema. O estágio acabou por correr bem e houve a oportunidade de integrar numa equipa e num projeto real, onde se foi muitas vezes confrontado com os problemas reais com que as equipas de consultores se deparam. Este tipo de participação não seria possível adquirir no meio académico, pelo que proporcionou uma experiência extremamente útil para enfrentar o mundo empresarial no futuro, desta vez já num emprego a sério e fora do âmbito de estágio.

A passagem por dois projetos, em dois clientes diferentes, no espaço de um mês foi igualmente um desafio pois foi preciso captar muito rapidamente a lógica de negócio, de forma a ser possível contribuir para os desenvolvimentos da equipa. E conseguir aprender de forma rápida e eficaz a lógica de negócio de dois projetos de grandes dimensões, e sobre um tema completamente novo, não foi fácil. Contudo, o facto de serem dois clientes tão diferentes e as fases dos projetos serem tão distintas – relembrando que no primeiro cliente o projeto estava em fase de análise e levantamento de requisitos; ao passo que, no segundo cliente, o projeto já se encontrava em fase de desenvolvimento – permitiu enriquecer o conhecimento a nível funcional assim como a nível técnico de um sistema de CRM. Estas duas passagens tornaram a adaptação no projeto principal muito mais fácil e produtiva.

No projeto principal, uma dificuldade inicial foi acompanhar o ritmo de trabalho da restante equipa. Naturalmente, os ritmos não eram iguais e havia sempre a pressão de não ficar para trás. A nível da parte funcional do projeto, nas reuniões com o cliente nem sempre foi possível reter tudo o que era discutido devido à falta de experiência neste tipo de ambiente. Contudo, e visto que pertencia a uma equipa, esta falta de experiência foi colmatada pela experiência da restante equipa. A componente documental do projeto foi igualmente desafiante, na medida em que não havia experiência prévia naquele tipo de documentação o que levou, várias vezes, à reescrita das secções já concluídas.

Na parte do desenvolvimento técnico, existiram dois obstáculos principais: o desenvolvimento de recursos web (em JavaScript) não pela linguagem em si, que já era bem conhecida devido à experiência académica, mas sim pela sua integração no contexto do CRM. O desconhecimento das bibliotecas utilizadas neste contexto também se tornou numa dificuldade, mas que, com o passar dos dias, foi ultrapassada. A grande dificuldade foi o desenvolvimento de plugins – linguagem C# - visto ser uma linguagem pouco ou quase nada abordada na faculdade e com nuances bastante distintas do JavaScript, por exemplo. O número de plugins desenvolvidos foi substancialmente inferior ao número de recursos web, o que também não ajudou a ultrapassar com tanta facilidade a dificuldade do desenvolvimento de *plugins*. Foi necessário ler a documentação e fazer um pouco de investigação sobre o assunto. No fim, tudo foi concluído com sucesso e foram atingidos os objetivos delineados.

## 4.4 Trabalho Futuro

Como trabalho futuro, seria interessante se o Dynamics 365 integrasse com um serviço que permitisse enviar comunicações para os telemóveis – via SMS – da mesma maneira que já tem um serviço integrado (com o *Outlook*) que permite aos utilizadores enviar mails aos seus clientes, diretamente do sistema do CRM. Seria um excelente aperfeiçoamento com muita utilidade às empresas, visto que esta integração acaba sempre por ser efetuada por meios externos ao CRM.

Uma funcionalidade que ainda não existe, *out-of-the-box*, mas que acaba quase sempre por ser implementada nas soluções das empresas, é a possibilidade de criar modelos de mails usando atributos de entidades customizadas e introduzindo-os, no modelo de mail, de forma dinâmica. Hoje em dia, apenas existe esta possibilidade para algumas entidades nativas do CRM e para os seus atributos nativos. Para conseguir atingir este fim, os programadores têm de recorrer a ferramentas externas ao CRM.

Relativamente ao bloco do *Customer Service*, seria interessante se as futuras versões do Dynamics 365 integrassem um sistema de comunicações via telefone, de forma a haver ligação direta entre a interação dos utilizadores com os clientes e com a plataforma do CRM. Esta funcionalidade dá uma maior agilidade às equipas de *Customer Service* e ajuda-os igualmente a ter um melhor desempenho no seu trabalho do quotidiano.

De realçar a importância de criar e manter toda a documentação dos projetos atualizada, pois por vezes as estratégias de implementação alteram, mas não são documentadas - retirando a coerência entre a documentação e o projeto. Uma documentação atualizada ajuda também na integração de novos membros nas equipas, pois a qualquer momento podem sair ou entrar novos membros na equipa.



## Referências Bibliográficas

- [1] Allan Pihl Pedersen, “*Microsoft Social Engagement: What is it?*”, Fev 11, 2016  
<http://alfapeople.com/microsoft-social-engagement-what-is-it/>  
(Acedido em 12-3-2017)
- [2] Archi, “*Overview*”  
<http://archi.cetis.ac.uk/>  
(Acedido em 21-1-2017)
- [3] Balsamiq, “*Balsamiq*”  
<https://balsamiq.com/>  
(Acedido em 1-08-2017)
- [4] Beal, V., “*CRM - Customer Relationship Management*”  
<http://www.webopedia.com/TERM/C/CRM.html>  
(Acedido em 12-1-2017)
- [5] Blue Note, “*CRM Cloud Computing*”, Jan 14, 2013  
<https://www.bluenote-systems.eu/crm-services/crm-cloud-computing.html>  
(Acedido em 24-09-2017)
- [6] Colman, E., “*What's the difference between Saas, Paas, Iaas?*”, Ago 27, 2013  
<https://www.computenext.com/blog/when-to-use-saas-paas-and-iaas/>  
(Acedido em 10-3-2017)
- [7] CRM Asia Solutions, “*What is CRM?*”  
<http://www.crmasiasolutions.com/online-business-solutions-articles/6-crm-concepts/76-what-is-crm-basic-concepts-of-crm-solutions.html>  
(Acedido em 21-1-2017)
- [8] Experienceux, “*What is Wireframing?*”  
<http://www.experienceux.co.uk/faqs/what-is-wireframing/>  
(Acedido em 1-08-2017)
- [9] Goswami, K., “*How would you explain IaaS, PaaS, SaaS to your Mom?*”, Ago 21, 2015  
<https://www.linkedin.com/pulse/how-would-you-explain-iaas-paas-saas-your-mom-kadamb-goswami/>  
(Acedido em 23-09-2017)
- [10] InterDyn BMI, “*What is Microsoft Dynamics CRM?*”




- <http://www.interdynbmi.com/microsoft-dynamics-crm>  
(Acedido em 21-1-2017)
- [11] Investopedia, "*Capital Expenditure (CAPEX)*"  
<http://www.investopedia.com/terms/c/capitalexpenditure.asp>  
(Acedido em 10-3-2017)
- [12] Investopedia, "*Europe, Middle East and Africa - EMEA*"  
<http://www.investopedia.com/terms/e/emea.asp>  
(Acedido em 13-3-2017)
- [13] Javlin, "*ETL (Extract-Transform-Load)*"  
<http://www.dataintegration.info/etl>  
(Acedido em 22-1-2017)
- [14] Manheim, S., Reizinixc, Taubensee, J., "*Service Bus queues, topics, and subscriptions*", chapter of Microsoft Azure Service Bus Documentation, Microsoft, Dez 12, 2016  
<https://docs.microsoft.com/pt-pt/azure/service-bus-messaging/service-bus-queues-topics-subscriptions>  
(Acedido em 12-3-2017)
- [15] MetaOption LLC, "*Out-of-the-box and Outside the Box: Two Sides of CRM for Business Management!*", Mai 18, 2015  
<https://community.dynamics.com/crm/b/metaoptioncrmblog/archive/2015/05/18/out-of-the-box-and-outside-the-box-two-sides-of-crm-for-business-management>  
(Acedido em 11-3-2017)
- [16] Michael M., "*Computação na cloud: Web-Based Applications that Change the Way You Work and Collaborate Online* (1 ed.)", Que Publishing Company, 2008
- [17] Microsoft, "*Introduction to entities in Microsoft D365*", Nov 29, 2016  
<https://msdn.microsoft.com/en-us/library/gg309396.aspx>  
(Acedido em 11-3-2017)
- [18] Microsoft, "*O que é o Team Foundation Server?*"  
<https://www.visualstudio.com/pt-br/tfs/>  
(Acedido em 16-1-2017)
- [19] Microsoft, "*SQL Server Integration Services*"  
<https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services>  
(Acedido em 31-07-2017)
- [20] Microsoft, "*SQL Server Management Studio*"

- <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms>  
(Acedido em 31-07-2017)
- [21] Microsoft, “*Register and Deploy Plug-Ins*”  
[https://msdn.microsoft.com/en-us/library/gg309620.aspx#bkmk\\_pluginregistration](https://msdn.microsoft.com/en-us/library/gg309620.aspx#bkmk_pluginregistration)  
(Acedido em 31-07-2017)
- [22] Microsoft, “*Use FetchXML to construct a query*”, Nov 29, 2016  
<https://msdn.microsoft.com/en-us/library/gg328117.aspx>  
(Acedido em 13-3-2017)
- [23] Microsoft, “*Visio*”  
<https://products.office.com/pt-pt/visio/flowchart-software>  
(Acedido em 22-1-2017)
- [24] Microsoft, “*Visual Studio*”  
<https://www.visualstudio.com>  
(Acedido em 21-1-2017)
- [25] Microsoft, “*What is Azure*”  
<https://azure.microsoft.com/pt-pt/overview/what-is-azure/>  
(Acedido em 22-1-2017)
- [26] Microsoft, “*What is Computação na cloud?*”  
<https://azure.microsoft.com/en-us/overview/what-is-Cloud-computing/>  
(Acedido em 12-1-2017)
- [27] NetSuite Inc., “*What is ERP?*”  
<http://www.netsuite.com/portal/resource/articles/erp/what-is-erp.shtml>  
(Acedido em 21-1-2017)
- [28] Oracle, “*Procedure Declaration and Definition*”  
[https://docs.oracle.com/cd/B28359\\_01/appdev.111/b28370/procedure.htm#LNPLS01336](https://docs.oracle.com/cd/B28359_01/appdev.111/b28370/procedure.htm#LNPLS01336)  
(Acedido em 31-07-2017)
- [29] Rackspace Support, “*Understanding the Computação na cloud Stack: SaaS, PaaS, IaaS*”, Jul 14, 2016  
<https://support.rackspace.com/how-to/understanding-the-cloud-computing-stack-saas-paas-iaas/>  
(Acedido em 12-1-2017)
- [30] Rouse, M., “*Data Lake*”, Mai 2015  
<http://searchaws.techtarget.com/definition/data-lake>

- (Acedido em 10-3-2017)
- [31] Salesforce, “*What is CRM?*”  
<https://www.salesforce.com/eu/crm/what-is-crm.jsp>  
(Acedido em 12-1-2017)
- [32] The CRM Book, “*Business Process Flows*”  
<http://crmbook.powerobjects.com/basics/business-process-flows/>  
(Acedido em 11-3-2017)
- [33] The CRM Book, “*Business Units*”  
<http://crmbook.powerobjects.com/system-administration/business-administration/business-units/>  
(Acedido em 11-3-2017)
- [34] Touzard, T., “*XrmToolBox*”  
<http://www.xrmtoolbox.com/>  
(Acedido em 5-2-2017)
- [35] Tutorials Point, “*Microsoft Dynamics CRM - Workflow Processes*”  
[https://www.tutorialspoint.com/microsoft\\_crm/microsoft\\_crm\\_workflow\\_processes.htm](https://www.tutorialspoint.com/microsoft_crm/microsoft_crm_workflow_processes.htm)  
(Acedido em 11-3-2017)
- [36] Tutorials Point, “*Understanding CRM Functional Modules*”  
[https://www.tutorialspoint.com/microsoft\\_crm/microsoft\\_crm\\_functional\\_modules.htm](https://www.tutorialspoint.com/microsoft_crm/microsoft_crm_functional_modules.htm)  
(Acedido em 15-1-2017)
- [37] Web Finance Inc., “*Streamline*”  
<http://www.businessdictionary.com/definition/streamline.html>  
(Acedido em 10-3-2017)

# Anexos

## Legenda:

- Linha normal 
- Tracejado pequeno 
- Tracejado grande 

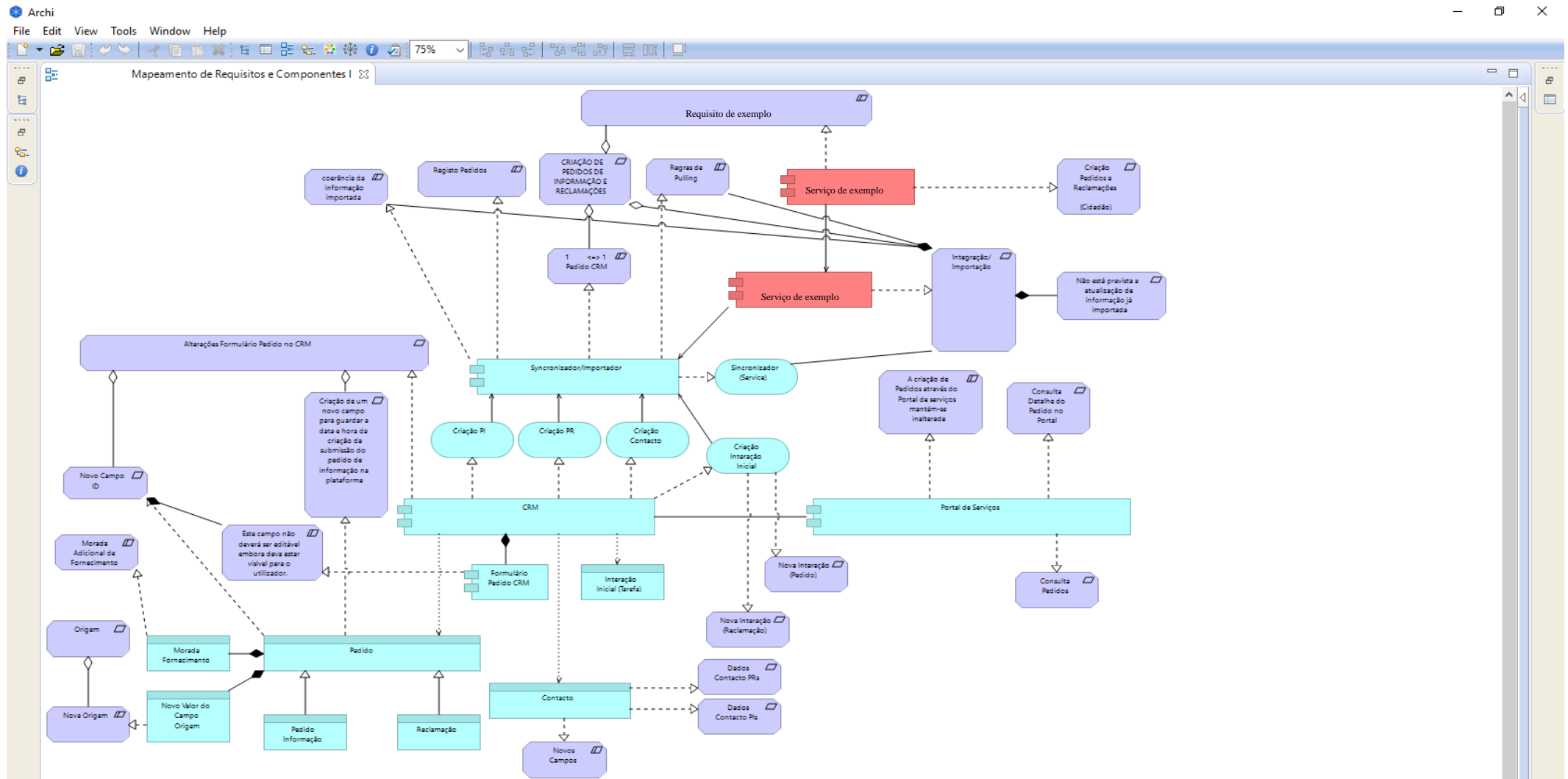
## **Anexo I. Organização da Equipa de Desenvolvimento de CRM (para B2C e B2B)**

Gestor de Projeto (B2C)				
Francisco Ceia				
Equipa de Migração de Dados	Equipa de Desenvolvimento		Equipa Funcional (Documentação)	
Tiago Santos	João Correia	Joana Pinto	João Correia	Joana Pinto

Gestor de Projeto (B2B)				
Luís Rosendo				
Equipa de Migração de Dados	Equipa de Desenvolvimento			Equipa Funcional (Documentação)
Tiago Santos	Joana Pinto	Pedro Correia	Joana Pinto	Tiago Santos
		Pedro Correia	Joana Pinto	



## Anexo II. *Archi*: Exemplo de Utilização no Projeto

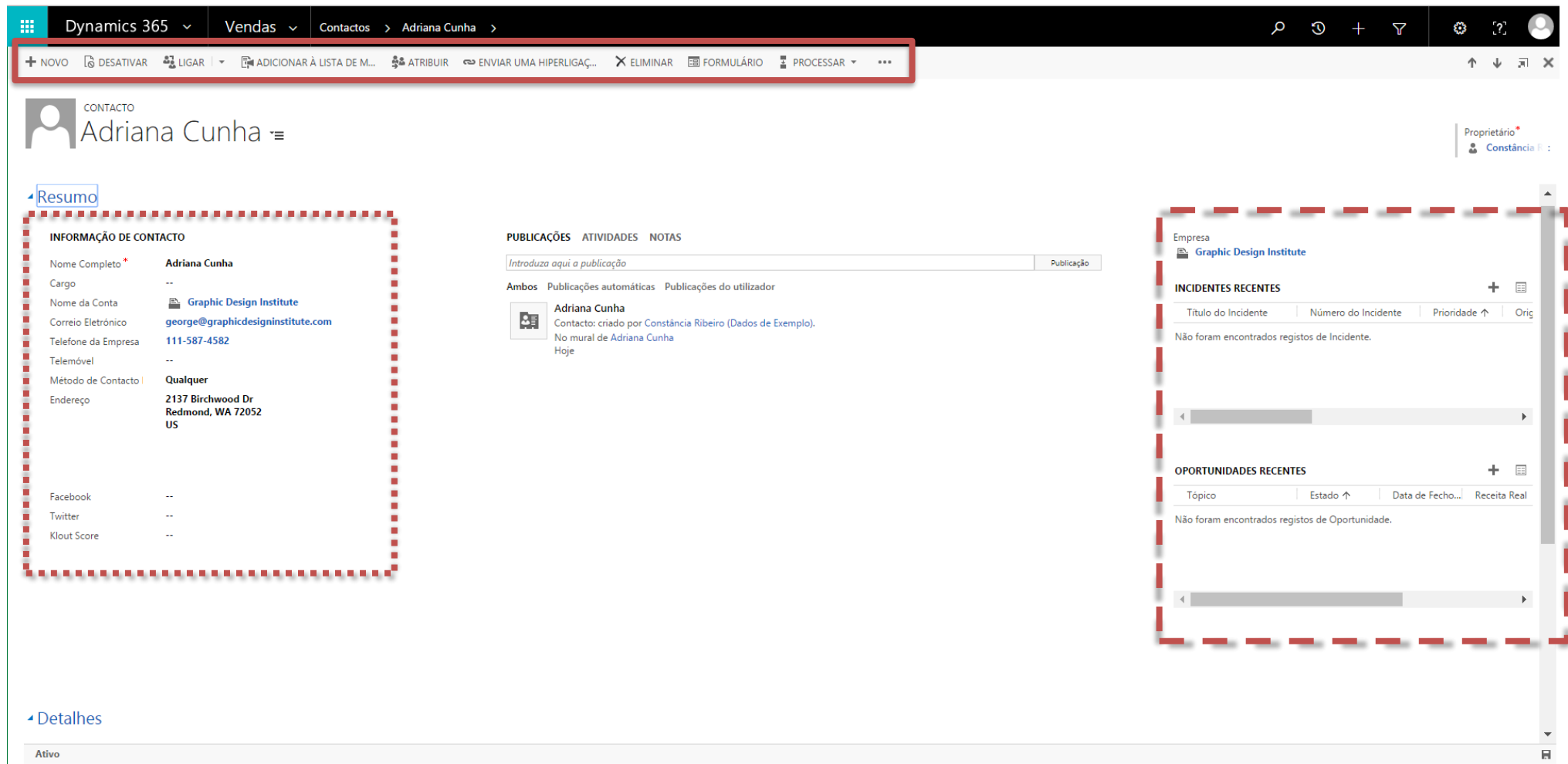






## Anexo III. Formulários Desenvolvidos para o Negócio B2C

Entidade “Contacto”



**Dynamics 365** | **Vendas** | **Contactos** > **Adriana Cunha**

**Resumo**

**INFORMAÇÃO DE CONTACTO**

Nome Completo *	Adriana Cunha
Cargo	--
Nome da Conta	Graphic Design Institute
Correio Eletrónico	george@graphicdesigninstitute.com
Telefone da Empresa	111-587-4582
Telemóvel	--
Método de Contacto	Qualquer
Endereço	2137 Birchwood Dr Redmond, WA 72052 US
Facebook	--
Twitter	--
Klout Score	--

**Publicações** | **ATIVIDADES** | **NOTAS**

*Introduza aqui a publicação* Publicação

**Adriana Cunha**  
Contacto: criado por *Constância Ribeiro* (Dados de Exemplo).  
No mural de *Adriana Cunha*  
Hoje

**Incidentes Recentes**

Título do Incidente	Número do Incidente	Prioridade ↑	Orig
Não foram encontrados registos de Incidente.			

**Oportunidades Recentes**

Tópico	Estado ↑	Data de Fecho...	Receita Real
Não foram encontrados registos de Oportunidade.			

**Detalhes**

Ativo

- Linha normal – seção de botões de ação
- Tracejado pequeno – informação sobre entidades relacionadas
- Tracejado grande – informação pessoal do contacto

Entidade “*Ticket*”

Customer Care

Tickets > ENC Nº 700010

SANDBOX

Está disponível uma nova versão do dashboard de Conhecimentos da Organização. Veja como este dashboard pode ajudá-lo a fomentar a adoção dos utilizadores finais e a evitar problemas de suporte.

Experimente-o agora

+ NOVO

REATIVAR INCIDENTE

PROCESSAR

MODELOS DO WORD

EXECUTAR RELATÓRIO

EDITOR DE FORMULÁRIOS

TICKET : INCIDENTE

ENC Nº 700010

Resumo

Detalhes do Ticket

Fã

1E8F62050A6C5301D05D64AA470BA846

Título do Ticket

ENC Nº 700010

Nº da Ocorrência

2017-002425

Tipo de Ticket

Operação

Canal de Origem

Portal

Departamento de Origem

Contact Center

Departamento Responsável

Contact Center

Assunto

Bilhética

Descrição

O sr. efectuou a compra de 5 bilhetes online com o nº de enc 700 010 , estando a mesma "PROCESSADA" desde dia 23/08/2016 sem que no entanto o adepto tenha recebido ainda o PDF .

Podem analisar por favor ?

TLF:

E-MAIL :

Obrigada,

Criado Por

CRM Service

Modificado Por

CRM Service

Criado Em

25/08/2016 12:07

Estado

Resolvido

Data de Resolução

26/08/2016

Nº de dias de Resolução

0

Ações

ATIVIDADES

NOTAS

Concluído Por CRM Service 18/04/2017 17:44

Detalhes do Fã

Resolvido

- Linha normal – informação do *Ticket*
- Tracejado pequeno – botões de ação sobre o *Ticket*
- Tracejado grande – interações registadas, com o cliente, no contexto do Ticket: mail trocados, chamadas, entre outros

## Formulário do Fã no Customer Care

 FÃ : CUSTOMER CARE

Francisco Correia

### ◀ Dados Pessoais


#### Informação do Fã

Nome do Cartão --  
Tipologia Adepto  
Categoria Não Definido  
Nome \* Francisco Correia  
Correio Eletrónico --  
Fonte Multimédia  
Cobrador Não Definido  
Valor Quotas em Dívida 0,00€

#### Outra Informação


Nome Casa do Benfica Não Definido  
Nº de Sócio de Casa do Benfica --  
Sócio Especial Não Definido

#### Campanhas


+ 			
Nome ↑	Tipo de Campanha	Código da Ca...	Razão do Esta...
Não foram encontrados registos de Campanha.			

Nº de Sócio --  
Data de Admissão 08/02/2017  
Telefone --  
Telemóvel 123456789  
Profissão --  
Situação Não Aplicável  
Quota Modalidades Não  
Última Quota Paga --

#### Agregado Familiar

		
Ligado A	Função (A) ↑	Descrição
Não foram encontrados registos de Ligação.		

#### Tickets

+ 				
Estado	Nº da Ocorrência	Título do Ticket ↑	Assunto	Departamento de Origem
Não foram encontrados registos de Ticket.				

#### Atividades

##### ATIVIDADES NOTAS

Tudo	▼	Adicionar Chamad...	Adicionar Tarefa	***	↓  
Não foi encontrado nenhum registo de atividade.					

Ativo



## Anexo IV. Menu de Criação e Configuração de um Formulário

Microsoft | Dynamics 365

Joana Pinto teste

**Explorador de Campos**

Filtrar: Todos os Campos

☒ Mostrar apenas campos não utilizados

- Alcunha
- Apellido
- Apellido Yomi
- Apenas Marketing
- Assistente
- Atividade Seguir Correo Eletrónico
- Câmbio
- Cliente de Back Office
- Criado Em
- Criado Por
- Criado Por (Delegado)
- Criado Por (Entidade Externa)
- Departamento
- Dia Preferencial
- Dimensão do Cliente
- Educação
- Empregado
- Endereço 1: Apartado
- Endereço 1: Código Postal
- Endereço 1: Concelho
- Endereço 1: Desvio UTC
- Endereço 1: Distrito
- Endereço 1: Fax
- Endereço 1: Latitude
- Endereço 1: Localidade
- Endereço 1: Longitude

**Novo Campo**

**Solução: Solução Predefinida**

**Formulário: Contacto**

**Resumo**

**INFORMAÇÃO DE CONTACTO**

Nome Completo: Nome Completo

Cargo: Cargo

Nome da Conta: Nome da Empresa

Correio Eletrónico: Correio Eletrónico

Telefone da Empresa: Telefone da Empresa

Telemóvel: Telemóvel

Método de Contacto Preferencial: Método de Contacto Preferencial

Endereço: Endereço 1

**Social Details**

Facebook: Facebook

Twitter: Twitter

Klout Score: Klout Score

Vista de Mapa

**PAINEL DE REDE SOCIAL**

**DETALHES DO CLIENTE**

Empresa: Nome da Empresa

Nome da Empresa

**INCIDENTES RECENTES**

INCIDENTES RECENTES

**OPORTUNIDADES RECENTES**

OPORTUNIDADES RECENTES

**Comum**

- Atividades
- Elegibilidades
- Perfis de Rede Social
- Ligações
- Histórico de Auditoria
- Comentários
- Listas de Preços de Co...

**Vendas**

- Oportunidades

**Serviço**

- Incidentes

**Marketing**

- Listas de Marketing

**Sessões de Processo**

**Contato**

- Resumo
- Detalhes

**Barra de Ferramentas:**

- Guardar
- Guardar e Fechar
- Publicar
- Alterar Propriedades
- Remover
- Anular
- Refazer
- Corpo
- Cabeçalho
- Rodapé
- Navegação
- Regras de Negócio
- Propriedades do Formulário
- Pré-visualização
- Ativar Direitos de Acesso
- Mostrar Dependências
- Propriedades Geridas
- Intercalar Formulários
- Atualizar

- Linha normal – campos da entidade disponíveis para colocar no formulário
- Tracejado pequeno – área onde se desenha o formulário
- Tracejado grande – seção de botões de configuração



## Anexo V. Construção de Regra de Negócio

Neste cenário em particular, esta regra de negócio foi criada de forma a bloquear determinados campos na entidade *Ticket* após a sua criação. Esta regra era necessária pois não era suposto os utilizadores poderem alterar informação no *Ticket* após a sua criação.

Guardar Como DESATIVAR

REGRAS DE NEGÓCIO: Ticket

**Campos de leitura após a criação do Ticket**

Clique para adicionar uma descrição

Adicionar Cortar Copiar Colar Eliminar Instantâneo

Condição Ticket criado?

Bloquear/Desbloquear Bloqueia campo "Fã"

Bloquear/Desbloquear Bloqueia campo "Título do Ticket"

Bloquear/Desbloquear Bloqueia campo "Tipo do Ticket"

Bloquear/Desbloquear Desbloqueia campo "Título do Ti..."

Bloquear/Desbloquear Desbloqueia campo "Tipo do Tic..."

Bloquear/Desbloquear Desbloqueia campo "Canal de C...

Regra de Negócio (Vista de Texto)

SE Criado Em contém dados

ENTÃO

- Bloquear Fã
- Bloquear Título do Ticket
- Bloquear Tipo de Ticket

Componentes Propriedades

Entidade Ticket

Regras + Novo

Regra 1

Origem Entidade

Campo Criado Em

Operador Contém dados

Expressão de Condição (Vista de Texto)

(Criado Em Contém dados)

Aplicar Eliminar

- Linha normal – ações caso passem (ou não) na condição
- Tracejado pequeno – menu onde se define as condições (onde se seleciona os campos)
- Tracejado grande – condições da regra de negócio

## Anexo VI. Construção de Fluxo de Trabalho

Geral Administração Notas

**▼ Ocultar Propriedades do Processo**

Nome do Processo

Ativar Como

**Disponível para Execução**

☐ Executar este fluxo de trabalho em segundo plano (recomendado)

☐ Como processo a pedido

☐ Como um processo subordinado

**Opções para Processos Automáticos**

Âmbito

Iniciar quando:  ☐ O registo foi criado

☐ Registrar alterações de estado

☐ O registo foi atribuído

☒ Registrar alteração de campos [Ver](#)

☐ O registo foi eliminado

Executar como: ☐ Proprietário do fluxo de trabalho

☒ Utilizador que fez alterações ao registo

Entidade

Categoria

**Retenção do Registo do Fluxo de Trabalho**

☒ Manter registos para tarefas de fluxo de trabalho com erro

- Linha normal – definição do modo em que o fluxo de trabalho deve ser acionado: de forma síncrona ou assíncrona
- Tracejado grande – definição das condições em que o fluxo de trabalho deve ser acionado

## Anexo VII. Formulários Desenvolvidos para o Negócio B2B

Entidade “Empresa” com Tipologia “Casa”



### Empresa

#### Empresa

Nome \*

Empresa de Teste

Grupo



Parceiro



Corporate



Prospect



Distribuidor



Casa



Evento



Fornecedor



Arrendatário



Patrocínio



Publicidade



Licenciamento



### Casa

#### Informação Principal

Casa

#### Inauguração

### Casa - Composição dos Órgãos Sociais

#### Direção

Órgão Social - Direção



SPORT CLUBE - Direção



## [Casa](#) - Composição dos Órgãos Sociais

### Direção

Órgão Social - Direção



SPORT CLUBE - Direção

Nome Presidente	<b>Paulo Henriques Medeiros</b>	Email Presidente	--
Nº de Sócio Presidente	--	Telefone Presidente	--
Nome Vice-Presidente	--	Email Vice-Presidente 1	--
Nº de Sócio Vice-Presidente 1	--	Telefone Vice-Presidente 1	--
Nome Vice-Presidente 2	--	Email Vice-Presidente 2	--
Nº de Sócio Vice-Presidente 2	--	Telefone Vice-Presidente 2	--
Nome Vice-Presidente 3	--	Email Vice-Presidente 3	--
Nº de Sócio Vice-Presidente 3	--	Telefone Vice-Presidente 3	--
Nome Vice-Presidente 4	--	Email Vice-Presidente 4	--
Nº de Sócio Vice-Presidente 4	--	Telefone Vice-Presidente 4	--
Nome Vice-Presidente 5	--	Email Vice-Presidente 5	--
Nº de Sócio Vice-Presidente 5	--	Telefone Vice-Presidente 5	--
Nome Vice-Presidente 6	--	Email Vice-Presidente 6	--
Nº de Sócio Vice-Presidente 6	--	Telefone Vice-Presidente 6	--





**Assembleia Geral**

Órgão Social - Assembl

**SPORT CLUBE - Assembleia Geral**

Nome Presidente	--
Nº Sócio Presidente	--
Nome Vice-Presidente	--
Nº de Sócio Vice-Presidente	--
Nome Primeiro Secretário	--
Nº Sócio Primeiro Secretário	--
Nome Segundo Secretário	--
Nº Sócio Segundo Secretário	--

Email Presidente	--
Telefone Presidente	--
Email Vice-Presidente	--
Telefone Vice-Presidente	--
Email Primeiro Secretário	--
Telefone Primeiro Secretário	--
Email Segundo Secretário	--
Telefone Segundo Secretário	--

**Conselho Fiscal**

Órgão Social - Consel

**SPORT CLUBE - Conselho Fiscal**

Nome Presidente	--
Nº Sócio Presidente	--
Nome Relator	--
Nº Sócio Relator	--
Nome Secretário	--

Email Presidente	--
Telefone Presidente	--
Email Relator	--
Telefone Relator	--
Email Secretário	--



Empresa

Nome\*

Grupo

Distribuidor

Arrendatário

Empresa de Teste

Parceiro

Casa

Patrocínio

Corporate

Evento

Publicidade

Prospect

Fornecedor

Licenciamento

Geral

Informação Principal

Número de Empresa --

Telefone Principal --

Correio Eletrónico --

Fax --

Web site --

Morada

Morada --

Localidade --

Código Postal --

Sub Código Postal --

Localidade CP --

Distrito/Provincia --

País --

Tipo de Cliente --

Resumo

PERFIL

Setor --

NIPC --

Capital Social --

Faturação Anual --

Hiperligação ao LinkedIn --

Observações --

N.º de Empregados --

C. Registo Comercial --

Data Fundação da Empresa --

Grupo Empresarial --

Marca --

Tem Relação Com a Fundação? Não